Everyday mapping: The design, construction and calibration of a portable surveying device

Terry Scott
St Hilda's College



Supervisor:
Professor Paul Newman

Department of Engineering Science
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy

30th January 2017

Terry Scott
St Hilda's College

Doctor of Philosophy
January 2017

Everyday mapping: The design, construction and calibration of a portable surveying device

Abstract

The contributions of this thesis can be categorised into two main areas. (1) The design, construction and testing of a portable mapping device, and (2) the automatic cross-calibration of cameras with laser scanners.

The mapping device, whose entire chassis is manufactured using a 3D printer, is designed to produce high quality 3D maps using a suite of cameras and laser scanners. A key feature of the unit is that it allows for multiple sensors configurations, thus enabling it as a versatile research tool. At the time of writing, the model produced for this thesis was in regular use by the Oxford Robotics Institute and a commercial company.

The second component of this thesis is partly an outgrowth of (1); replacing or repositioning sensors on the device requires recalibration of those sensors. We propose and provide analysis for two automatic methods of calibration which use observations of the robot's surroundings to register a camera and laser. The first method uses the information theoretic measure *normalised information distance*, which is found to perform well but only under certain scene conditions. This leads to our proposal of a scene selection policy which ensures only effective scenes are used for the registration. The second camera–laser calibration method relies on the automatic extraction and matching of image point features between modalities.

Lastly, we propose a unifying calibration framework that formulates multiple sensor registrations as a pose graph constrained by known inter-sensor measurements. Using an alternating optimisation we are able to treat each expensive registration (as part of a global optimisation) independently.

Calibration methods are tested in diverse environments of on-road driving, warehouse and urban pedestrian sidewalks. We demonstrate successful, accurate calibration in each case.

Declaration of Authorship

This thesis is submitted to the Department of Engineering Science, University of Oxford, in fulfilment of the requirements for the degree of Doctor of Philosophy. This thesis is entirely my own work, and except where otherwise stated, describes my own research.

Terry Scott, St Hilda's College

Acknowledgements

First and foremost, I would like to express my utmost gratitude to my supervisor, Paul Newman. For his support, the odd game of cricket, and opening me to a world of opportunity. My time spent with the exceptional people at MRG (now the ORI) has been equal parts intimidating, exhilarating and rewarding. My co-authors Lina, Akshay and Pedro require special thanks for their invaluable time, expertise and good humour.

I must also thank friends and family, old and new, for their boundless support and friendship. They are possibly my greatest fortune.

To my family

Contents

1 Introduction **1**
- 1.1 Everyday mapping . . . 2
- 1.2 A framework for automatic camera–laser calibration . . . 4
  - 1.2.1 Calibrating a camera and laser scanner . . . 4
  - 1.2.2 Selecting reliable scenes for calibration . . . 5
  - 1.2.3 Unifying a network of sensors . . . 6
  - 1.2.4 Publications . . . 6

2 Portable Mapping **7**
- 2.1 Introducing *NABU 3* . . . 10
  - 2.1.1 Functional specification . . . 11
- 2.2 Design and construction . . . 13
  - 2.2.1 Sensors . . . 13
  - 2.2.2 Chassis . . . 15
  - 2.2.3 Sensor mounting . . . 16
  - 2.2.4 Assembly . . . 18
  - 2.2.5 Electronics . . . 20
- 2.3 Operation and testing . . . 21
  - 2.3.1 User interface . . . 21
  - 2.3.2 Deployment . . . 22
  - 2.3.3 Configurations . . . 23
- 2.4 Concluding discussion and possible future development . . . 24
  - 2.4.1 Calibration of sensors . . . 25
  - 2.4.2 3D printing for development . . . 26

3 Platforms and Datasets **28**
- 3.1 Overview of sensors . . . 28
- 3.2 Nissan LEAF RobotCar . . . 30
  - 3.2.1 Oxford 10km road dataset . . . 32
- 3.3 GEM Survey Vehicle . . . 32
  - 3.3.1 Milton Keynes dataset . . . 34
- 3.4 *NABU* surveying device . . . 35
  - 3.4.1 Warehouse dataset . . . 36

4 Background **37**
4.1 Related Work . . . 38
4.1.1 Calibration targets . . . 38
4.1.2 Correspondence selection in natural scenes . . . 40
4.1.3 Global registration in natural scenes . . . 41
4.1.4 Registration using information metrics . . . 43
4.2 Operations and conventions in $\mathbf{SE}(3)$ . . . 44
4.2.1 Notation . . . 46
4.3 Point cloud swathe generation . . . 46
4.4 Camera and projection models . . . 48
4.4.1 Creating a synthetic image of a scene . . . 49
4.5 Information Theory . . . 50
4.5.1 Entropy . . . 50
4.5.2 Mutual Information . . . 52
4.5.3 Normalisation . . . 53
4.6 Nonlinear Optimisation . . . 55
4.6.1 Gauss–Newton nonlinear least squares . . . 56
4.6.2 Nelder-Mead (simplex) . . . 58
4.7 Evaluating calibration . . . 60
4.7.1 Measuring calibration performance . . . 61
4.7.2 Reporting angles . . . 62
4.7.3 'Ground truth' calibration using a motion tracker . . . 63
4.8 Concluding remarks . . . 65

5 Camera–Laser Calibration **68**
5.1 Calibration by minimising Normalised Information Distance . . . 69
5.1.1 Method . . . 70
5.1.2 Data pre-processing . . . 72
5.1.3 Results: NID . . . 77
5.1.4 Discussion . . . 81
5.2 Calibration by automatic 2D point feature matching . . . 85
5.2.1 Method . . . 86
5.2.2 Results: SIFT point features . . . 88
5.2.3 Discussion . . . 93
5.3 Conclusions and future work . . . 94
5.3.1 NID calibration . . . 96
5.3.2 Auto point feature calibration . . . 98

6 Intelligent Scene Selection **99**
6.1 Effects of scene variation . . . 101
6.2 Scene selection via cost-surface analysis . . . 103
6.2.1 Verifying a potential minimum . . . 103
6.2.2 Global minimum via SVM classification . . . 104
6.2.3 Scene scoring based on cost-basin shape . . . 107
6.3 Concluding remarks . . . 112

6.4 Future work . . . 115
6.4.1 Scene selection based on image features . . . 116
6.4.2 Discussion on future work . . . 118

7 Unified calibration by pose graph optimisation 120
7.1 Method . . . 121
7.1.1 Notation . . . 123
7.1.2 Graph structure . . . 123
7.1.3 Global objective function formulation . . . 125
7.1.4 Decoupling the objective function . . . 126
7.2 Results . . . 131
7.3 Conclusions . . . 134
7.3.1 Future work . . . 137

8 Conclusion and future work 139
8.1 Summary of contributions . . . 139
8.2 Future work . . . 143
8.2.1 Portable Mapping . . . 143
8.2.2 Calibration . . . 144

Bibliography 148

List of Figures

1.1 Coloured 3D map of a warehouse section 2
1.2 Visualisation of the camera–laser calibration problem 3

2.1 Computer rendering of first and second generation *NABU* 8
2.2 'Zebedee' and Riegl mapping devices 9
2.3 Computer render of *NABU 3* with overall dimensions 10
2.4 *NABU 3* main chassis exploded view 15
2.5 Stereo-camera unit exploded view 17
2.6 Sensor–chassis mounting interface 18
2.7 Lidar unit installation on main chassis 19
2.8 *NABU 3* exploded view of chassis and sensor attachments 20
2.9 Mission Control – web interface for data acquisition 21
2.10 Close-up of *NABU 3* sensor payload 22
2.11 *NABU 3* use cases: Rail and Warehouse surveying 23
2.12 Coloured 3D map of a railway section 24

3.1 3D survey creation using a camera and laser 30
3.2 Diagram of the Nissan LEAF showing sensor configuration 31
3.3 Satellite map of Oxford 10km road dataset and examples 31
3.4 Sensor setup of the GEM survey vehicle 33
3.5 Example camera images from Milton Keynes dataset 34
3.6 Computer rendering of *NABU 3* with two laser scanners 35
3.7 Warehouse used for the *NABU 3* dataset 35

4.1 Setup of the calibration problem presented in [93] 39
4.2 Camera and laser images created using [52]. 42
4.3 Notational conventions for transforms in $\mathbf{SE}(3)$ 46
4.4 Creating a 3D point cloud swathe 47
4.5 Camera image and sparse synthetic laser image. 49
4.6 Venn diagram of bi-variate Entropy. 50
4.7 Examples of distributions and their entropy. 51
4.8 Image registration example 52
4.9 Joint entropy measures varying with the distribution 54
4.10 Cost surfaces of the image registration example 54
4.11 NID cost surface and its derivative 58

4.12 Expansion, reflection and contraction of Nelder-Mead simplex 59
4.13 Graph of convergence for Nelder-Mead algorithm 60
4.14 Calibrated vs. Uncalibrated image overlay 61
4.15 A visual guide for understanding von Mises-Fisher distribution 63
4.16 Obtaining 'ground truth' calibration using motion tracker 64

5.1 Laser–Camera point projection and histogram generation. 72
5.2 Camera and laser raw data distributions 73
5.3 Camera and laser data distributions post-equalisation 73
5.4 Histogram of an image before and after equalisation 74
5.5 Camera and synthetic laser image after Gaussian smoothing 75
5.6 Effects of Gaussian smoothing of camera and laser data on the NID . 79
5.7 Adding more scenes to the optimisation 81
5.8 Possible source of uncertainty due to laser configuration 84
5.9 SIFT features matched between camera and laser image. 86
5.10 Point cloud segment of a warehouse generated by *NABU 3* 89
5.11 Graphs of camera–laser calibration error, *NABU 3* 91
5.12 Graphs of camera–laser calibration error, GEM 92
5.13 Error histograms: uncalibrated vs calibrated (point features) 93
5.14 Computation time for calibration algorithms 95

6.1 Scene variation and cost surface convexity 100
6.2 Camera image and anomalous laser image 102
6.3 Creating a scene descriptor by sampling the cost function 105
6.4 Labelled scene descriptors 106
6.5 Precision–recall of scene classification 107
6.6 'Smoothness' and 'distinctness' criteria for scene selection 109
6.7 Mean squared error of random vs selected scenes 111
6.8 Example high–scoring scenes 112
6.9 Example low–scoring scenes 113
6.10 Plot of solutions using *random* vs. *selected* scenes 114
6.11 Distribution of scene scores for an Oxford and warehouse dataset . . 115
6.12 Adding a checkerboard to a bare scene 116
6.13 Precision–recall classification using *Bag of Words* and GIST features 118

7.1 Example sensor network with notation 122
7.2 Constructing the sensor pose graph 124
7.3 Graph of the sensor network, accounting for ego-motion 129
7.4 Error plot for alternating calibration - stereo camera 132
7.5 Error plot for alternating calibration - mono camera 132
7.6 Sensor setup of the GEM survey vehicle 133
7.7 Cost surface evolution for the lower-level objective function 134
7.8 Convergence of lower- and upper-level solutions 135
7.9 Laser–camera overlay, comparison of visual alignment 136

List of Tables

2.1 List of sensors used on *NABU 3* 14

3.1 List of sensors and their descriptions 29
3.2 Sensors used on the LEAF platform 32
3.3 Sensors used on the GEM platform 33
3.4 Sensors used on the *NABU 3* platform 36

5.1 Calibration Results (NID): GEM Bumblebee–Laser1 78
5.2 Calibration Results (NID): LEAF Bumblebee–Laser1 78
5.3 Calibration Results (NID): LEAF Bumblebee–Laser2 79
5.4 Calibration Results (NID): *NABU 3* Bumblebee–Laser1 80
5.5 Calibration results (point features): *NABU 3* Bumblebee–Laser1 . . . 90
5.6 Calibration results (point features): GEM Bumblebee–Laser1 90

List of Terms

NABU **Na**vigation **B**ase **U**nit. The range of portable mapping devices used by the the ORI.

NABU 3 The third generation *NABU* device, a contribution of this thesis.

NID Normalised Information Distance. An information theoretic metric which measures the difference in joint information contained in two distributions.

ODT Object Detection and Tracking. The process of identifying obstacles, cars, pedestrians etc., and tracking their motion.

ORI Oxford Robotics Institute. Formerly, Mobile Robotics Group.

push-broom A particular configuration of planar laser scanner that sets the scanning plane orthogonal to the direction of motion. This allows subsequent scans to be appended according to a known ego-motion, thereby generating a 3D point cloud.

SLAM Simultaneous Localisation and Mapping. A core system of a mobile robot which records and identifies landmarks landmarks observed by its sensors.

VO Visual Odometry. The determination of ego-motion, using consecutive frames of either a mono or stereo camera.

List of Symbols

x Scalar value, non-bold and italicised.

$\mathbf{x}$ Vector or tuple, small bold.

$\mathbf{X}$ Matrix, capitalised bold.

$\hat{\mathbf{x}}$ Hatted variable, estimated value.

$\bar{\mathbf{x}}$ Barred variable, measured or otherwise known constant.

$\oplus, \ominus$ Right and left composition operation between two $\mathbf{SE}(3)$ groups.

$\mathbf{g}$ Standard symbol used to denote pose in $\mathfrak{se}(3)$.

$\mathbf{R}(\cdot)$ Rotation exponential map operation, $\mathbf{R} : \mathfrak{so}(3) \to \mathbf{SO}(3)_{3\times3}$.

$G(\cdot)$ Transform map exponential operation, $G : \mathfrak{se}(3) \to \mathbf{SE}(3)_{4\times4}$.

Chapter 1

Introduction

Mobile robots across applications operate under common principles of navigation, perception and planning. We follow the philosophy of 'anywhere autonomy', which is to say that the fundamental software and hardware systems of a mobile robot should be broadly transferable between robots, environments and tasks. This thesis aims to advance the capabilities of mobile robots in this regard in two main ways.

The first is in the design, construction and testing of a portable mapping device, called the *NABU 3*. Robots frequently rely on detailed prior maps of their environment to localise, perceive and plan. The *NABU 3* aims to be a fully-featured and versatile development unit, allowing for the easy creation of these maps.

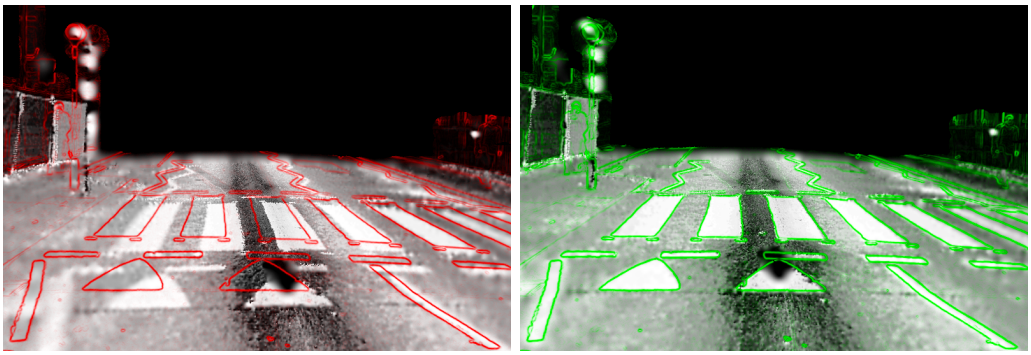
The second research problem arises not only as a prerequisite for a robot's operation, but as a consequence of the *NABU 3*. The task of *cross-calibrating* multiple cameras and laser scanners is compounded by the *NABU 3*'s allowing for the reconfiguring of its sensors. The remainder of this thesis thus addresses the problem and challenges associated with automatically obtaining and maintaining accurate sensor calibration over a robot's life. The following introductory sections discuss the motivation for and provide an overview of the contributions made in this thesis.



Figure 1.1: Coloured 3D map of a warehouse section generated using the *NABU 3* mounted to a forklift. The colour is acquired from each 3D laser point's associated camera pixel, which affirms the importance of accurate camera–laser calibration.

1.1 Everyday mapping

A prior map is a set of points, features or landmarks which are spatially related, and which have been recorded in a stand-alone survey operation. The coloured point cloud in Figure 1.1 offers an example of a map we intend to create. As well as to inform a robot about its surroundings, the map can be used for research purposes. For example, the abundance of real-world data can be used to test localisation systems [3, 5, 55], or to supply machine learning algorithms for object detection and tracking (ODT) [87]. Detailed 3D surveys also have application beyond autonomous navigation and research; they aid planning and design for architects, civil engineers, on road or rail, or in mines, to give but a few examples.



Uncalibrated Calibrated

Figure 1.2: Visualisation of the camera–laser calibration problem. In both cases, an edge-filtered camera image of a pedestrian crossing is superimposed on an image constructed using a laser point cloud. The alignment of the two modalities is predicated on accurate inter-sensor pose, the solving of which is the problem of *calibration*. The left and right images show the subsequent sensor alignment of an uncalibrated and well-calibrated system respectively.

A core contribution of this thesis is the design of a device that allows users to easily record detailed surveys almost anywhere, thus directly facilitating research. In Chapter 2 we introduce the *NABU 3*, the third generation portable mapping device developed at the Oxford Robotics Institute (ORI). We discuss the design and motivations in detail, and contrast this version with others previously developed for research and industry.

The *NABU 3* offers portable fully-featured 3D mapping, as the total unit weight is under 9kg. A key feature is the reconfigurable sensor mounting interface, made possible by the use of 3D printing. Sensors can be repositioned or replaced depending on the desired task, thus making the device a versatile research tool.

The maps generated by the *NABU* require the fusion of camera and laser information. The relative pose between these sensors therefore must be precisely known to produce metrically accurate maps. This is the problem of *calibration*, which is concisely represented by Figure 1.2. The sensors must also be recalibrated each time the configuration is adjusted, providing motivation for a fully automatic approach to calibration, and a segue to the second contribution of this thesis.

1.2 A framework for automatic camera–laser calibration

The challenge of calibration is not unique to the *NABU*. Rather, a robotic system consisting of multiple sensors requires their relative poses to be known accurately. A practical approach for mobile robots is to fuse the active sensing of the laser scanner with the passive detection of a camera for reliable object tracking [27, 68]. Accurate inter-sensor calibration therefore ensures correct data association between multiple modalities. A policy of online, automatic calibration would be particularly useful to monitor and correct for sensors which fall out of alignment through use. While this particular use case is not covered in the scope of this thesis, we acknowledge it as a significant motivator.

This thesis focuses on the automatic cross-calibration of cameras and planar laser scanners. The overarching aim is to develop a framework that can be used to reliably obtain and maintain accurate calibration over the life of a robot. Chapter 4 provides a literature review, preliminary information and sets the context for the proceeding chapters.

1.2.1 Calibrating a camera and laser scanner

Historically, calibration has been achieved using target patterns simultaneously observable by both sensors [84, 93]. More recently, automatic approaches to calibration have emerged which eschew the need for calibration targets, and use implicit features of the observed world to register camera and laser data [41, 52, 62]. We present and analyse two approaches to camera–laser calibration, which are covered in Chapter 5.

Information theoretic measures for camera–laser registration

Measures of information such as *mutual information* have been used extensively in medical image registration [65] and visual servoing [20], and are well suited to comparing differing sensor modalities. In Section 5.1 we reformulate the localisation problem of [76] by applying the information-theoretic measure of *normalised information distance* (NID) to the registration of camera luminance to laser reflectance. Similar work has been done in [62], which uses mutual information to register a camera to a 3D Velodyne scanner. We find the resulting cost surface to be non-smooth, thus opt for the gradient-free optimisation method of Nelder-Mead.

Image features from a synthetic laser image

Calibration methods that use explicit feature correspondences between the two modalities typically either provide patterns for easy feature recognition [36], or require that correspondences be selected by hand [28]. The method proposed in Section 5.2 draws from the science of photogrammetry and its methods of registering a camera with a high-fidelity terrestrial laser scanner [14, 50]. Using a common image feature extractor, we automatically find and match features present in both the camera image and synthetic laser image. This is achieved using the Hokuyo UTM-30LX, a low-cost planar laser scanner. Registration then involves minimising the squared-error between the pixels' locations and their associated laser projections.

1.2.2 Selecting reliable scenes for calibration

The NID formulation is desirable since is typically robust to noise, does not require explicit feature matches, and is agnostic about the spatial relation between laser points. We do however discover in Section 5.1 that this formulation is not robust across all scene; some laser–camera observation pairs produce reliable calibration results while others cause failure. Other authors note that increasing the number

of observations improves the calibration, yet we argue that doing so is insufficient in some cases. We therefore propose a policy of *scene selection* in Chapter 6 that analyses the set of available scenes and only includes those that contribute positively to the registration.

1.2.3 Unifying a network of sensors

Lastly, having previously considered the calibration of a single camera–laser pair, we present an extension to calibration in Chapter 7 that provides a unifying framework for a network of sensors. In contrast to methods that formulate constraints from observations [26, 59], we treat the pairwise sensor registration as a computationally expensive black box. Instead we leverage known calibration between cameras, such as that of the well calibrated stereo camera unit, to constrain the pose graph.

Using an alternating optimisation, the parameter space is partitioned such that each pairwise sensor registration is treated independently from the pose graph optimisation. We test on a system of camera and laser pairs, yet the formulation offers the ability to incorporate any pair of sensors.

1.2.4 Publications

The work of Chapter 7 proposing the unified calibration framework has been published and presented The International Conference of Intelligent Robots and Systems in 2015 [71]. Also expanding on the calibration framework, we presented preliminary scene selection work of Chapter 6 at The International Conference of Robotics and Automation in 2016 [72].

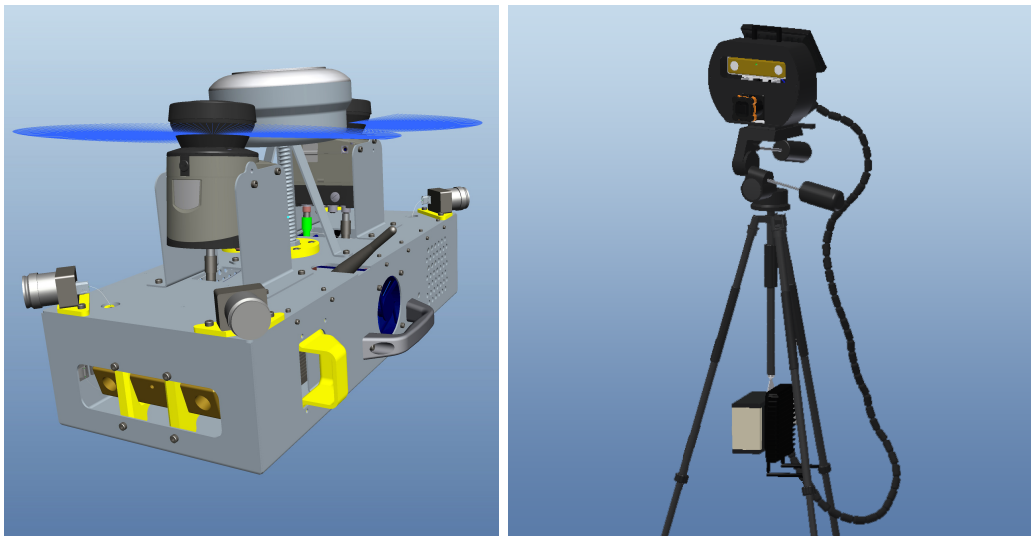
Chapter 2

Portable Mapping

We are for 'anywhere autonomy', which means that mobile autonomous technology should be applicable across contexts. The introduction of the robotic fleet in Chapter 3 demonstrates this; a robot can operate in a diverse range of settings. The detailed 3D maps against which a robot traversing the environment can reliably localise are key to this autonomy, and *portable mapping* allows us to generate these maps quickly and easily by having a single unit composed of all the necessary hardware and software.

The contribution of this chapter is the design, construction and testing of a new portable mapping device; the third generation in ORI's line. We build on the previous iterations of such a device, distilling the key elements of each. The result is a self-contained, modular, light-weight system with flexible options for mounting. The design is inspired by a few other notable systems which are worth mentioning here.

The first in ORI's range of portable mapping was designed by PhD student Ian Baldwin in 2010 [4], of which a computer generated model is available in Figure 2.1a. This original device housed four mono cameras, a stereo camera, two SICK LMS-151 2D laser scanners and a Novatel inertial navigation system (INS) aided by GPS (Global Positioning System). The frame was constructed almost entirely from aluminium due to being relatively strong, lightweight and easy to work with. This



a b

Figure 2.1: (a) Computer rendering of the first generation *NABU* — 'Naboo' — designed in 2010 by Ian Baldwin, a DPhil student at the time, and (b) the more compact *NABU 2* designed in 2013 by Kirsty McNaught, visitor to the research group.

was a fully-fledged system operating with high-fidelity lidar and odometry. Significant drawbacks were its lack of portability and difficulty of mounting; its weight and shape meant that it required a roof-rack or other suitably stable platform.

The second iteration condensed the most fundamental aspects of the first version (single 2D laser scanner, stereo camera and micro-INS unit) and produced a more portable hand-held version. The computer is a fanless unit boasting an Intel i7 processor, and is external to the rest of the device meaning mounting is unhindered by non-sensor components. Designed in 2013 by Kirsty McNaught, the housing makes use of 3D printing to manufacture otherwise complicated geometry. Figure 2.1b shows the device mounted on a tripod, with the computer connected by a tether.

We consider two other mapping devices, each seated at opposing ends of the complexity spectrum, namely the Zebedee designed by Bosse et al [15] and the industrial-grade Riegl VMX-450 [80]. The former is a hand-operated 'wobbling' 2D scanner and micro-INS, while the latter cannot even be lifted by hand — a high fidelity device capable of dense coloured 3D scene reconstructions. The ergonomics of these devices



Figure 2.2: The hand-held 'Zebedee' which sports a planar laser scanner and INS, left, and the large Riegl VMX-450 mounted to the roof-rack of an SUV.

are best described in an image, which we refer the reader to in Figure 2.2.

The 'Zebedee' is inspirational in its simplicity; the low-fidelity lidar with micro-INS provide basic functionality of 3D mapping at an affordable price. The INS and 2D scanner are mounted to the end of a spring, which provides the motion that is required by the scanner to capture all three dimensions of our world. The INS captures both the motion induced by the spring, and by the movement of the bearer, thus providing the ego-motion required to successfully generate the 3D map. The device itself may be hand-held, but the computational power typically sits in the backpack of the user.

In contrast, Riegl offers the pinnacle of mobile mapping solutions. The VMX-450 (predecessor to the newer VMX-AH1) houses two lidar scanners in a configuration similar to the 'push-broom' described earlier, albeit positioned such that the planes subtend the 'forward' direction by what appears to be about 45°. This effectively doubles the scanning rate and allows for scanning of surfaces orientated orthogonally to the vehicle trajectory. Multiple high-resolution cameras are placed around the unit to capture the appearance of the scene in full detail.

The remainder of this chapter is focused on the design, construction and operation of the *NABU 3*, the next-generation model of mapping devices produced by ORI.

We start by introducing the device, the motivation, features and intended purpose in Section 2.1. We also list the functional specifications the design will adhere to, while Section 2.2 covers the design in detail. The device is tested in Section 2.3, in which we

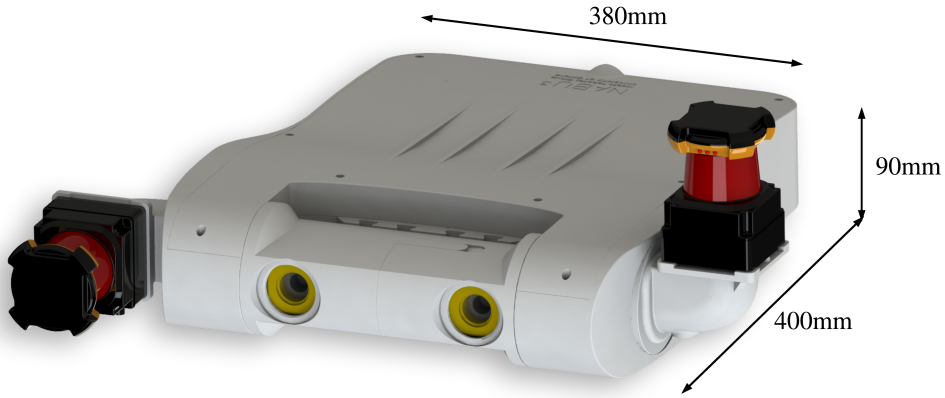


Figure 2.3: Computer render of *NABU 3* with overall dimensions. The device is configured with two planar laser scanners; horizontal for object detection and scan matching, and vertical for push-broom SLAM.

demonstrate its operation and typical usage scenarios. The chapter concludes with a discussion about the utility of the *NABU* and we make recommendations for future developments.

2.1 Introducing *NABU 3*

The third generation mapping device sits neatly between the first and second generations in terms of size, fidelity and portability. The versatility however has been notably increased; the modular sensor interface allows for bespoke configurations of lasers and cameras, particularly useful for researching optimal configurations for a given task. A computer render of *NABU 3* is shown in Figure 2.3.

The purpose of the *NABU 3* is to provide a highly portable mapping platform suitable for robotics research groups and industry that can provide high-fidelity maps at a low cost. A typical use case is for it to be mounted to a working vehicle to record surveys of the workplace. Vehicle behaviour is then naturally a component that the *NABU* records, valuable for researchers of autonomous planning and control.

The first task was to address the shortcomings of the previous generations. The intent was to have at least the functionality of *NABU 1* yet in a package that was

easy to handle and affix to different platforms. It could be argued that fidelity had to be sacrificed since we selected the smaller Hokuyo laser scanner over the larger SICK LMS-151, and a micro-INS/GPS over the Novatel system.

The second-generation *NABU* features a single binocular stereo camera, a Hokuyo 2D laser scanner in 'push-broom' configuration (used to generate 3D maps, as explained in Section 4.3), and a micro-INS/GPS. The computer is contained in a separate unit and is tethered to the sensor payload.

NABU 3 reverts to a self-contained system, housing multiple lasers and cameras, and a micro-INS/GPS. Lasers can be configured for dual push-broom for dense mapping, or single push-broom and horizontally for top-down *scan matching*. Odometry is provided by the stereo camera, with the option for INS assistance.

2.1.1 Functional specification

1. **Chassis**

   (a) The chassis will be sufficiently rigid as not to distort, which would put sensors out of calibration.

   (b) The device must be fully enclosed and offer protection from dust, debris, and light rain.

   (c) The device must be visually appealing and should be produced with high quality design, materials and finish.

   (d) The device will have mounting holes or other versatile interface capable of supporting the weight of the device.

2. **Sensors**

   (a) The unit will house a binocular stereo camera which has the ability to tilt at least $\pm 90^\circ$ from the 'straight ahead' position. Tilt must occur in stepped increments.

(b) The unit will be modular; two interfacing points will each offer ability to mount different configurations of either a single camera, or laser scanner. This interface has the following requirements:

- The interface will rigidly support up to two sensors.
- Must allow one degree of rotation of the attachment, through full 360°, in stepped increments.
- The attachment can be rotated and secured without the use of tools.

(c) The unit will house a GPS and INS.

3. **Computing and electrics**

(a) The unit (computer and sensors) will be powered by a single Li-Po battery pack.

(b) The battery must be removable without the use of tools (by hand).

(c) The unit will have the option to be powered externally using appropriate DC power connectors.

(d) The unit will house an on-board computer with the following requirements:

- Data ports to serve all sensors
- Wireless and wired networking
- Ability to connect monitor, mouse and keyboard.
- Ability to record raw data streams of all on-board sensors simultaneously.

(e) The chassis must be designed in such a way that all the internal components can be accessed and removed easily with a minimum number of non-specialist tools.

(f) It must be possible to connect an external display, keyboard, mouse and ethernet to the device with ease.

4. **Operation**

   (a) The device must be easily carried by one person without difficulty or assistance.

   (b) In normal operation, the user will interface with the device computer using a remote computer or tablet over ethernet (wired or wireless).

   (c) The device must be designed to withstand operation on an off-road vehicle at slow speeds.

   (d) Device can be operated in vertical or horizontal positions.

2.2 Design and construction

The capability of today's hardware is such that we can scale down the sensor payload and computer to create a new fully-fledged portable mapping device – one which can be carried by a user. In this section we will cover the design and reasons for particular elements of the new device. The advent of additive manufacturing has afforded us the design flexibility which would not be available to us by using traditional manufacturing methods. This is important as the unit is designed to be modular; it can be reconfigured and fitted with different sensors depending on the specific task it is performing. We demonstrate a possible configuration in Figure 2.3 whereby one lidar scans horizontally for object detection and scan matching, and the other vertically for push-broom localisation and mapping.

2.2.1 Sensors

The design hinges around a sensor configuration consisting of (at least) a binocular stereo camera and two planar laser scanner — we demonstrate in Section 2.3 that *NABU 3* is capable of accommodating two additional mono cameras.

Table 2.1 lists the sensors used on *NABU 3*.

Table 2.1: List of sensors used on *NABU 3*

Sensor	Model	Specifications
Stereo camera	Point Grey Bumblebee2	1024×768 @ 20Hz
Mono cameras	Point Grey Grasshopper	1384×1036 @ 23Hz
Laser scanner	Hokuyo UTM-30LX	270°, 43.2kS/s, 2400rpm
INS/GPS	MicroStrain 3DM-GX4-45	INS: $\pm 5g$, 300°/sec, ± 2.5 Gauss. GPS: ± 2.5m vert, ± 5m hor.

The **Point Grey Bumblebee2** offers precise inter-camera calibration, and it has been used in systems that demonstrate consistent and accurate visual odometry [18, 74]. The model chosen for the *NABU 3* interfaces using only a single FireWire 800 connection, reducing the requirement for additional power distribution or data interfaces.

The original *NABU* employed two SICK LMS-151 planar laser scanners, each weighing 1.1kg. The power draw (60W), size and weight meant that this scanner was an unsuitable candidate given the requirement for *NABU 3* to be human-portable and battery operated. We opted instead for the **Hokuyo UTM-30LX**, which is similar to the SICK in terms of range, accuracy and resolution. Weighing in at only 280g and rated at 8W, it is the superior choice for a portable outdoor-use system. We therefore ensure that we are at least matching the performance of NABU 2 by using the same scanner, but doubling the sample rate given that two scanners can be recording simultaneously. Multi-plane scanners were infeasible given their size and cost.

The addition of two **Point Grey Grasshopper** cameras coupled with wide angle lenses allow for the creation of colour-accurate 3D point clouds, and accurate vision-based localisation and mapping.

As a reference system, we choose the **MicroStrain 3DM-GX4-45** for its small size, high-resolution Inertial Measurement Unit and integrated GPS.

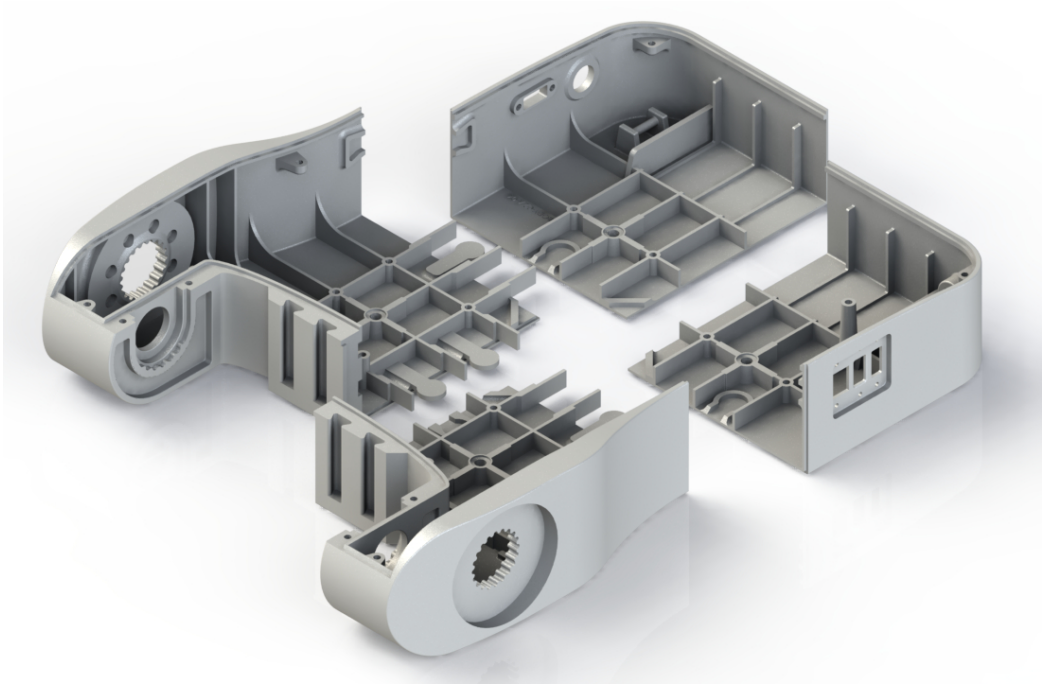


Figure 2.4: *NABU 3* main chassis exploded view. Due to the size of the device and the size limitations of 3D printing, the chassis had to be printed in parts and glued. The rendering shows the four parts of the chassis prior to assembly.

2.2.2 Chassis

The chassis is the main body of the device and has two main functions; to house and protect the internal electronic components, and to provide rigid mounting points for necessary sensors.

The *NABU 3* is also designed to be compact and light as it will be need to be capable of being operated by hand. In order to maximise design flexibility and space efficiency, the chassis is a bespoke design intended to be manufactured using an additive manufacturing method (3D printing). This method of manufacture is suitable due to the low-volume production, fast manufacture time and intricacy of the components. Each component needs not only to be designed with limitations of 3D printing in mind, but also to take advantage of the benefits it offers.

Material: The chosen material, an ultra-violet–curable acrylic plastic, does impose some constraints. The material is more brittle and has a lower tensile strength

than other common manufacturing plastics like ABS or nylon. It is therefore important to consider how this may impact the durability of the component.

Construction: The chassis has a unibody design that reduces the complexity and the number of fasteners needed for assembly. The size limitations of the 3D printer used meant that the chassis had to be printed in sections. Figure 2.4 shows a computer rendering of four parts of the chassis. Note the dovetails used to provide reliable and accurate locating of each section, while the edges are glued together. The large bay houses the computer and battery pack, with sufficient space for wiring. Panel mounting points for ethernet, HDMI and USB ports are included in the side of the chassis. It was noted that previous versions of the *NABU 3* lacked adequate connectivity options without having to access the device internally.

Strengthening: Since we are using a material which has a lower tensile strength than ABS, a plastic typically used for an application such is this, we over-design for strength. The ABS-type material for the 3D printer was unavailable at the time of manufacturing. The weight contribution of over-design is negligible compared to the weight added by the sensors, battery and computer. We therefore use a conservative body thickness of 3mm and add 2mm $\times$ 10mm ribs across the base of the chassis.

Mounting points: There are six holes in the base of the unit which are fitted with M5 threaded brass inserts. These holes can be used to attach mounting plates or tripods.

2.2.3 Sensor mounting

The placement and mounting of the sensors is an intrinsic part of the chassis design. Placement of the sensors is crucial to ensure correct operation of the device. We also intend the device to be modular, meaning that the sensor placement can be adjusted, or that different sensors can be attached simply creating a sensor-specific bracket.

Stereo-camera: We opt for a Bumblebee2 stereo camera to provide reliable camera–camera calibration and visual odometry. The attachment for the camera is a

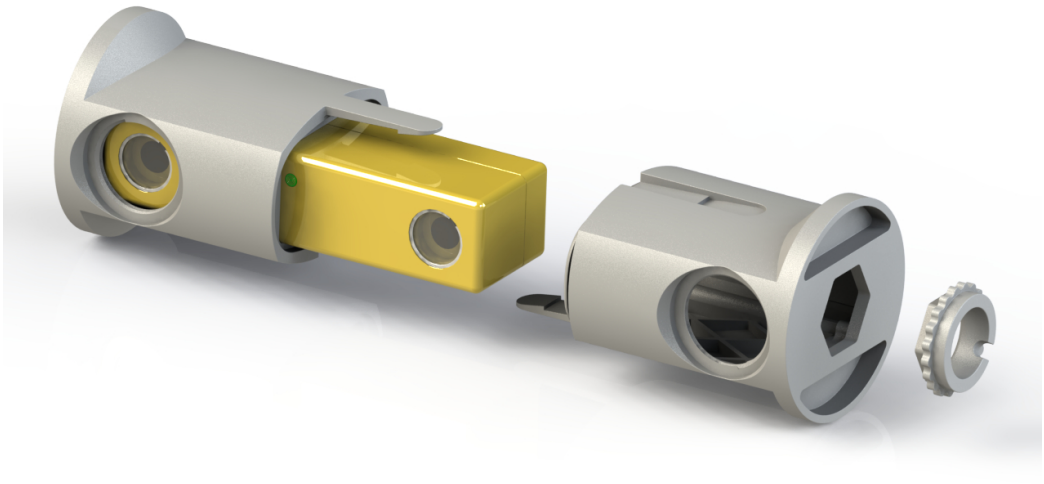
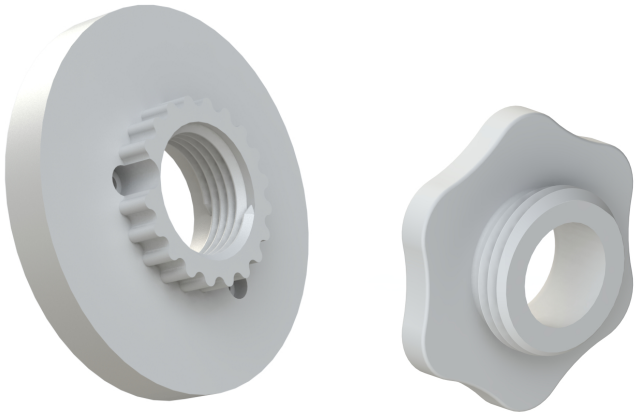


Figure 2.5: Stereo-camera unit exploded view. The unit is sealed to protect the camera against dust and splashes. The PointGray Bumblebee2 fits securely and assembly does not require tools. The splined piece installed into the end of the housing acts to secure the assembly to the chassis at a known angle.

sealed unit to guard against dust and splashes and can be assembled without tools. Figure 2.5 shows an exploded view of the camera assembly. The holes at either end allow for the pass-through of connecting cables, while the splined interfaces with the chassis to provide selectable angle increments of 18°.

Side sensor interface: On either side of the *NABU 3* chassis is a generic mounting point. This consists of a splined hole through which sensor wires can pass. The sensor attachment mates with this splined hole at an angle selected by the user, thus allowing for this adjustability. The spline allows for increments of 18° to be selected, and also allowing for easier calibration since the mounting angle will be known almost precisely. An angle encoder to detect position would alleviate the need to manually modify the extrinsic calibration scripts when a sensor is repositioned. While this indeed would be an advance towards end-to-end autonomy, we anticipate multiple configurations to be attached at this interface. The dual problem of detecting connected components and the angle at which the module is attached puts this solution out of scope for the project, given also that configurations would be altered infrequently.



Sensor mounting interface Securing screw

Figure 2.6: Sensor–chassis mounting interface. The left rendering shows the splined interface that mates with the corresponding spline on the chassis. The screw on the right has a thread that corresponds to the sensor interface, and is hollow to allow cable pass-through.

Each sensor attachment has a specifically designed screw thread with a corresponding screw. These coupling components are shown in Figure 2.6 where is is shown how the screw is designed such that sensor wires are able to pass through the centre. The screw is designed to be used by hand such the replacement of sensor attachments need not require any tooling. An exploded view showing the assembly order is given in Figure 2.7. This interface can be used to attach small laser scanners such as the Hokuyo, and monocular cameras.

Laser attachment: The laser scanner attachment uses the interface described above which uses the spline–screw mechanism. The attachment assembly is shown in Figure 2.7. The sensor's wires need to be passed through the attachment, chassis and securing screw before assembly. Note the splined interface from Figure 2.6.

2.2.4 Assembly

To keep assembly easy, we limited the use of fasteners to M3 socket head screws wherever possible. Reconfiguring the device should not require any additional tools

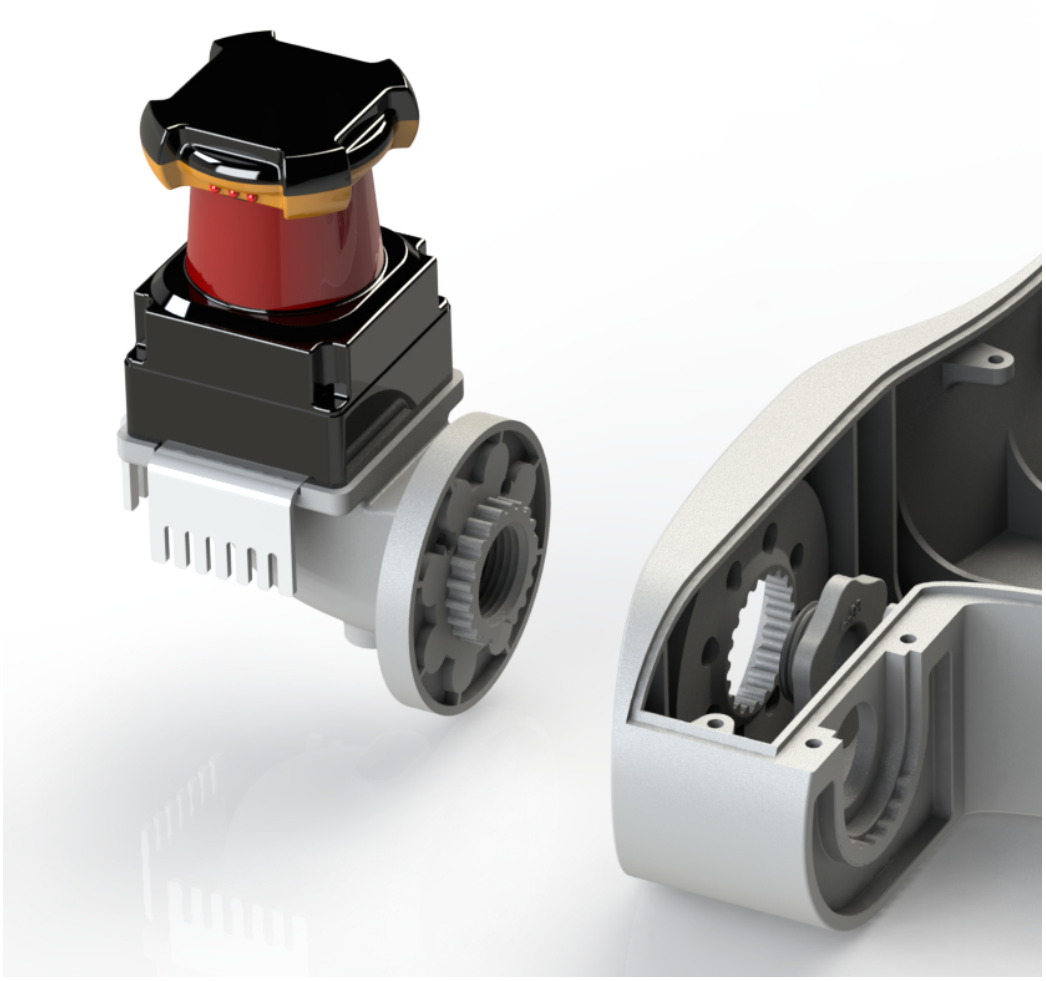


Figure 2.7: Lidar unit installation on main chassis. The mounting arm for the lidar is fixed to the main chassis using a specially designed screw. The angle that the arm can be mounted at is fixed using a spline, thus there is the ability to change the angle of the lidar in 18° increments.

and the side sensor attachments can be installed and removed without tools, although entry to the device through the main cover is required.

Figure 2.8 shows an exploded view of the assembly of the chassis and sensor attachments. The electronic components such as the computer and battery have been excluded here for clarity.

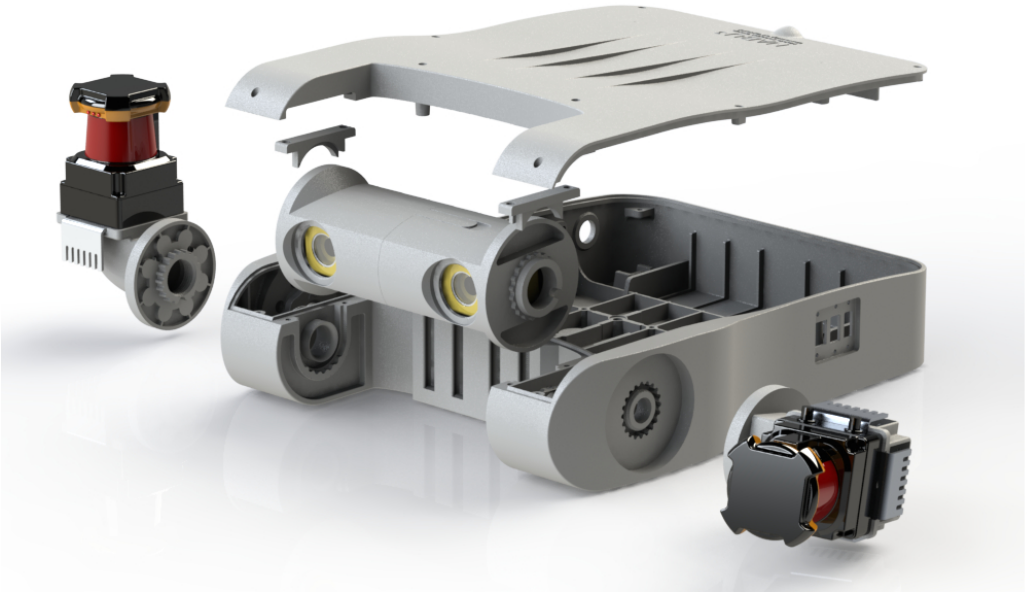
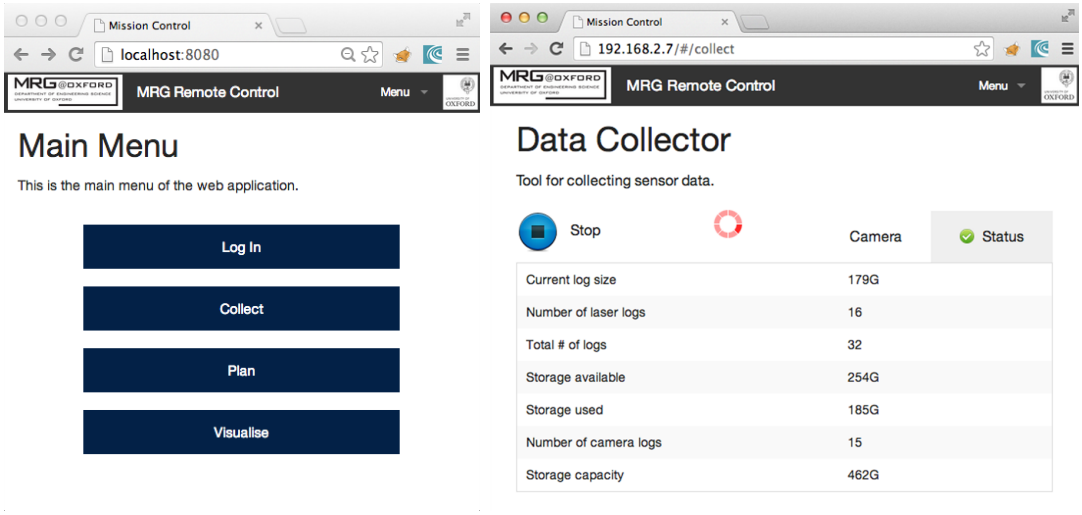


Figure 2.8: *NABU 3* exploded view of chassis and sensor attachments. The stereo-camera unit is inserted from the top of the device and secured with two clamps, each secured with two screws. The laser attachments are each secured with a specially designed screw that allows pass-through for cables.

2.2.5 Electronics

At the heart of *NABU 3* is a fanless PC running an Intel i7 quad-core processor. Its PCI expansion ports allow for an additional array of FireWire 800 ports required by the cameras. The solid-state hard disk is a necessary addition to accommodate the rate at which raw sensor data is recorded.

The 14.7V high-capacity Li-Po battery provides approximately 30 minutes of recording time. The battery is accessible via a hatch at the rear of the device and is secured using a hook-and-loop strap for easy removal. The voltage is regulated to 12V to supply the computer and laser scanners.



Web interface dashboard Data acquisition with real-time statistics

Figure 2.9: Mission Control – a web interface for data acquisition. The left-hand frame shows the dashboard for interacting with the NABU. Once the user has gained access, they can elect to start the data acquisition, for which the interface (right) updates the user on the system status.

2.3 Operation and testing

2.3.1 User interface

Operation of the *NABU 3* is accomplished using a remote computer. The Mission Control launcher is an easy-to-use user interface that, although developed by ORI and therefore not covered in this thesis, is important to mention as it is a crucial part of the whole system.

Switching the device on powers up the sensors and boots the PC into a state such that it can be remotely accessed over a private wireless network via PC or tablet. Mission Control provides the user with a graphical interface to begin and end a survey, with the option to visualise the camera frames as they are being collected. Each component of the system can be enabled or disabled depending on whether it is required for the particular operation. The web dashboard and data acquisition screens are presented in Figure 2.9. In this example, the size of recorded data for the

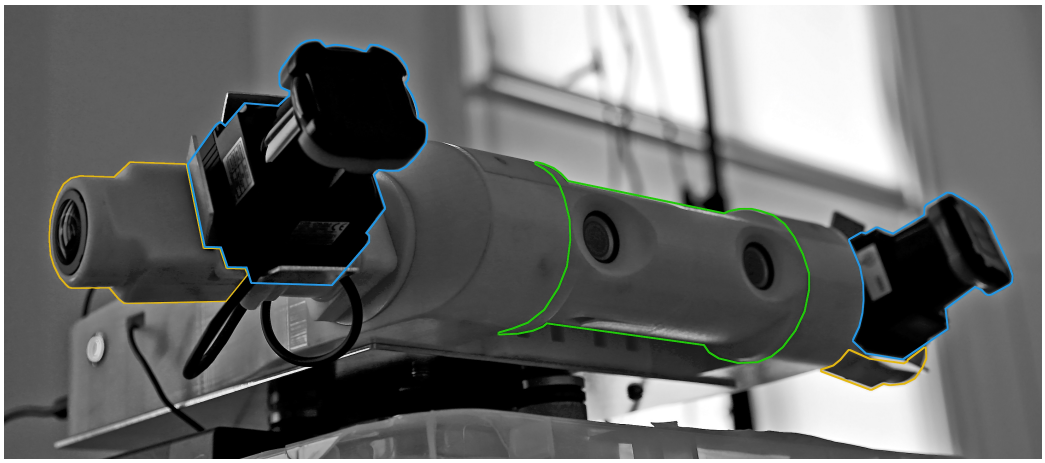


Figure 2.10: Close-up of *NABU 3* sensor payload. The mono-camera (yellow) and laser (blue) attachments are a custom design for the desired purpose of high-fidelity mapping. The stereo camera module (green) is in the default position.

current session totals 179GB, with 254GB of disk space remaining.

The web-based implementation allows for complete cross-platform functionality, provided the device has a web browser and wireless networking.

2.3.2 Deployment

The *NABU 3* has been deployed in a number of different cases thus demonstrating its intended usage and versatility. Of particular interest are cases where regular vehicles may not travel, or in confined spaces. The two scenarios which use to exemplify this are a railway and warehouse, respectively.

Rail: The *NABU* has been used for asset tracking on railways by mounting it to a maintenance vehicle shown in Figure 2.11. An example of the resulting survey is shown in Figure 2.12 and has been coloured using the side-mounted monocular cameras.

Warehouse: An interesting application of autonomous technology is robotic forklifts operating in warehouses. We mounted the *NABU* to a forklift, as shown in Figure 2.11, to generate detailed maps of a warehouse. A section of the resulting map is given in Figure 1.1. This particular figure makes it clear that a good camera–laser



Figure 2.11: *NABU 3* use cases: Rail and Warehouse surveying. The *NABU* is mounted to a railway maintenance vehicle (left) and a forklift using a high-quality consumer grade tripod.

calibration is required to ensure correct colour association of the laser points.

2.3.3 Configurations

An important design feature of the *NABU 3* is that it is modular and configurable. 3D printing has made it possible to design custom parts for particular sensors to interface with the chassis as in Figure 2.7, with a fast turnaround. Testing of the *NABU 3* was performed using the configuration shown in Figure 2.10; each sensor mount holds a camera and push-broom laser scanner.

The splined mounting locations have shown to be adequate for shifting the angles of the sensor mounts. There is no play or backlash as a locating clearance fit was chosen. The stereo camera unit in Figure 2.11a has been angled downward such that more of that railway and surrounding terrain is captured in the cameras' fields of view. Potentially awkward mounting positions can thus be foregone in favour of a more appropriate sensor configuration.

The *NABU* is able to operate in the vertical position due to the stereo camera module having full 360° motion in increments of 18°. The process of rotating the



Figure 2.12: Coloured 3D map of a railway section generated using the *NABU 3* mounted to a railway car.

camera module requires the removal of the main cover of the device, and two securing clamps. We also anticipate that the *NABU 3* can allow for more creative mounting locations, such being worn vertically as a backpack with adequate shoulder straps, although this has not been tested.

2.4 Concluding discussion and possible future development

This chapter detailed the design, construction and testing of a new surveying device for the robotics research group at Oxford. The device ultimately performed as designed and is still in regular service at the time of writing. The device has been used to demonstrate the research group's capabilities on road and rail, in warehouses and offices, and makes significant progress towards accessible, high-quality scene mapping and reconstruction. The device is a prototype and thus allows us make recommendations for improvements and future developments, which we do so here.

As a research unit, the *NABU 3* performs well. The modular interfaces allow for a broad range of experimentation using different sensors and placement. One device will suffice for a multitude of tasks, which is particularly important when there are cost constraints. The sensor modularity allows different configurations to be tested quickly and easily so they can be optimised; the dual push-broom configuration in Figure 2.10 has proven generally to be versatile and to provide high quality reconstructions.

While portable, the device is not intended to be operated by hand. The overall weight of a fully laden unit is approximately 8kg, which makes it relatively easy to carry but means the user may fatigue during survey recording. We would propose fitting a backpack–style carrying system that would attach directly to the mounting points at the base of the chassis. Ergonomic hand-held usage would require the device to be redesigned, with emphasis on the unit's weight and appropriate handholds.

The device fits between its two predecessors in terms of cost, weight and size. The chassis is manufactured using the relatively expensive 3D printing process, but this cost is offset by choosing more economical laser scanners. while larger and heavier than *NABU 2*, the self-contained unit is more versatile given the ease with which it can be transported and deployed.

2.4.1 Calibration of sensors

The *NABU 3*'s trademark features of modularity and reconfigurability do impose a significant challenge in obtaining and maintaining the correct inter-sensor calibration. We discovered that even dimensions supplied by the CAD software do not offer 'correct' calibration due to possible implicit bias in the sensors, and inability to measure a sensor's true centre.

The splined locators for the sensor interfaces provide convenient a priori estimates for calibration since the spline increments are of a known angle. Informing the software of this change remains an unsolved problem since a modification invariably causes the prior estimate to fall outside the basin of convergence of the optimisation

stage of calibrating. The new estimate would currently have to be entered manually.

The remainder of this thesis attempts to solve precisely the problem of calibrating a laser–camera sensor suite, and maintaining this calibration 'fully automatically'. We use the term tentatively since calibration tools (particularly for camera–laser) require an initial calibration estimate, often supplied by the user. The CAD software in this case does offer sufficiently close estimates to ensure convergence using our calibration algorithms.

2.4.2 3D printing for development

The design of *NABU 3* was predicated on its construction using 3D additive printing. The limitless geometric possibilities allowed for a bespoke and efficient design unattainable using traditional manufacturing. Plastic mould design requires an expert understanding of the limitations of overhangs, recesses and material thickness — the prohibitive cost for prototyping notwithstanding.

The fast production of modified sensor modules meant a fast turnaround for testing of different configurations. The *NABU 3* was designed to be a research unit; to be modifiable and thus evolve as the mapping and academic requirements change. In this regard, 3D printing technology has proven itself as an invaluable tool.

The technology is not without fault. Perhaps the most inhibiting factor is material strength — although this improves continuously with innovation. As mentioned in the *Design* section of this chapter, the acrylic material does not have the strength nor durability of ABS, thus the *NABU 3* chassis has been over-designed for strength.

The second factor that was taken into consideration was manufacturing tolerance and surface finish. The *NABU* was printed on an *Objet Eden PolyJet* machine, which boasts an accuracy of 20-85μm [78]. This accuracy came into consideration whilst designing the spline locators for the stereo camera and side sensor modules. The spline is designed to be a *locating clearance* fit, which would nominally allow easy assembly with near-absent backlash. The splined shaft–hole combination was designed with

moderate clearance (0.1mm) yet the parts assembled with difficulty. We suggest that due to a relatively poor and inconsistent surface finish (compared to machined parts), mated parts should be prototyped independently to account for characteristics of a particular printer. A relationship between *fit* and *clearance* can then be determined.

Chapter 3

Platforms and Datasets

This chapter introduces the fleet of robotic vehicles we use for testing, their sensors and intended use, and corresponding datasets. Automatic calibration must be platform-independent. We therefore test calibration on three different vehicles, each operating in different environments, scales and sensor configurations.

3.1 Overview of sensors

While the vehicles at ORI each typically host the basic recipe of at least a stereo camera and planar laser range finder, they are generally intended for different purposes. More specifically, each sensor is designed and placed to perform a particular service in the autonomy stack of *navigation*, *perception* and *planning*. Key to each of the sensor's functions is the interactive relationship it has with other sensors. For the purposes of this thesis, we are interested in cameras and planar (2D) laser scanners, for which we list and summarise the specifications for each in Table 3.1.

The stereo cameras offer an inexpensive and accurate solution to ego-motion estimation and serve predominantly as visual odometry (VO) source. Front and rear facing stereo cameras not only provide ego-motion and localisation [19, 45], but play a crucial role in perception [7, 55, 64, 83], as do the surrounding mono-cameras.

Table 3.1: List of sensors used on the robotics platforms, and their descriptions

	**Hokuyo UTM-30LX: planar laser scanner** A compact 2D scanner that interfaces via USB making it convenient for use on small robots. It offers long-range capability (30m), making it suitable for use indoors and outdoors. The spinning scanner operates at 50Hz, in a 270° scanning arc with angular resolution of 0.25°.
	**SICK LMS-151: planar laser scanner** A larger scanner which offers high outdoor performance and is rated IP67. The scanner operates at 25Hz or 50Hz, with an angular resolution of 0.25° or 0.5° respectively.
	**Point Grey Bumblebee2 and XB3: stereo camera** Captures two or three (XB3) streams of colour images at 20Hz over FireWire. Cameras are precisely calibrated, allowing for accurate ego-motion estimation using visual odometry.
	**Point Grey Grasshopper2: mono camera** Offers high framerate and resolution using a global shutter, 2.0MP at 29Hz over FireWire. Typically coupled with a wide-angle lens to provide a larger field of view for mapping and perception.

We focus on the configuration of stereo cameras and 2D laser scanners used for mapping. Figure 3.1 demonstrates the creation of a coloured 3D survey using the 'push-broom' method coined in [5]. Generating an accurate survey requires the extrinsic calibration of two subsystems; laser–to–egomotion, and egomotion–to–camera. These coloured maps can be used to localise a robot using only a camera on subsequent traversals [63, 77]. By adapting this localisation problem, we can reformulate it as a calibration problem, as covered in Chapter 5. Generation of these swathes is covered in Section 4.3.

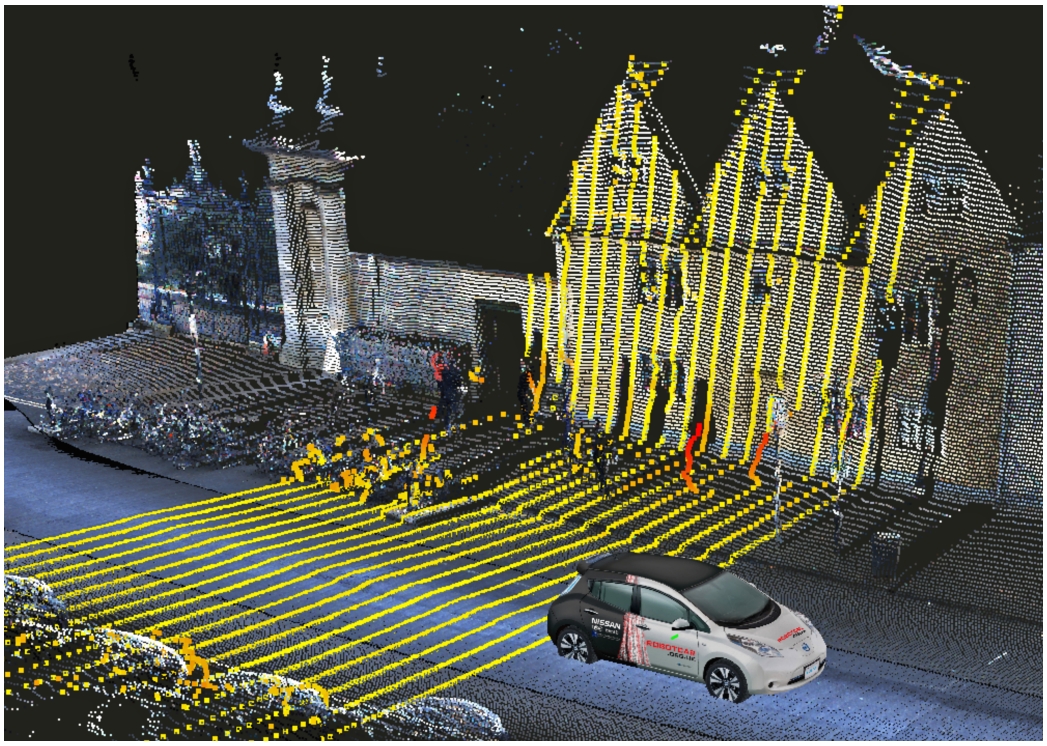


Figure 3.1: 3D survey creation using a camera and laser. A 3D swathe is generated using the 'push-broom' method, and the resulting points are coloured using the surrounding cameras. The yellow points are illustrative of sequential planar scans.

3.2 Nissan LEAF RobotCar

ORI's current flagship autonomous vehicle is the Nissan LEAF RobotCar. This electric vehicle relies on two SICK 2D laser scanners, a multi-planar scanner, surround mono-cameras and XB3 stereo camera to provide autonomy without moving or obtrusive sensors.

Again, navigation relies on vision and the generation of 3D swathes using push-broom lasers. The two planar laser scanners are positioned at the front and rear bumpers of the vehicle respectively, as illustrated in Figure 3.2. A summary of relevant sensors on this vehicle along with their specifications is offered in Table 3.2.

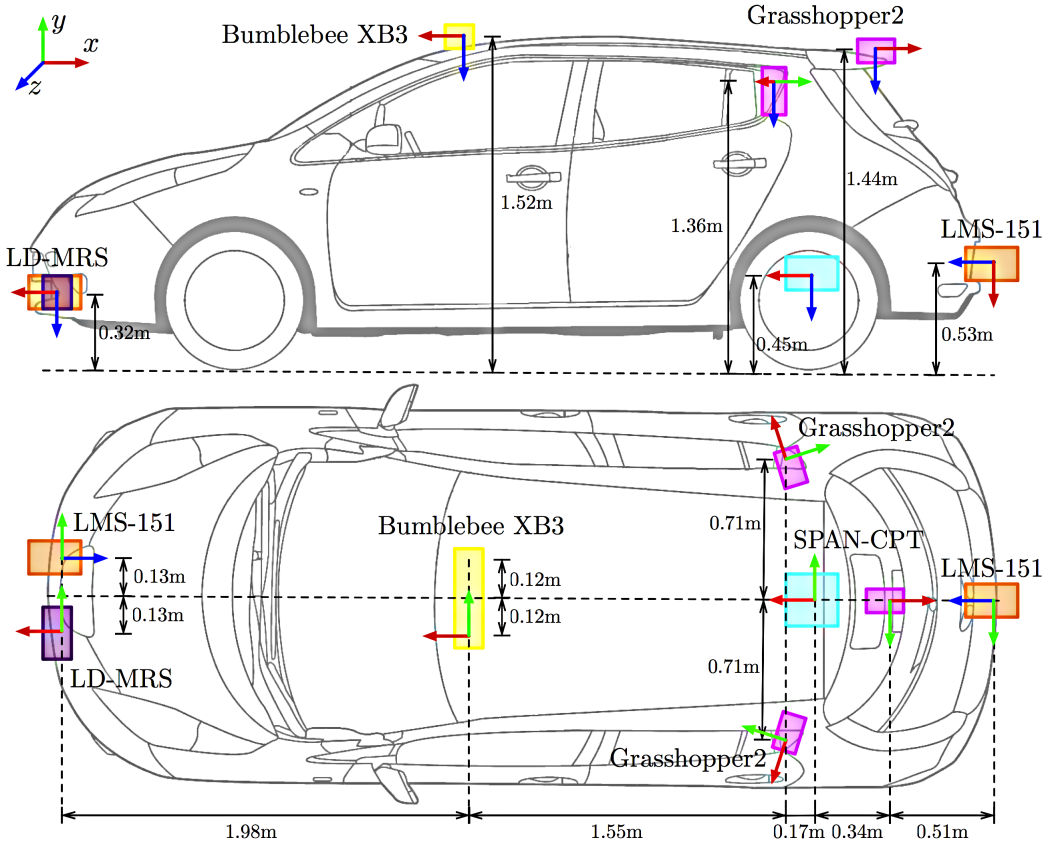


Figure 3.2: Diagram of the Nissan LEAF showing sensor configuration.

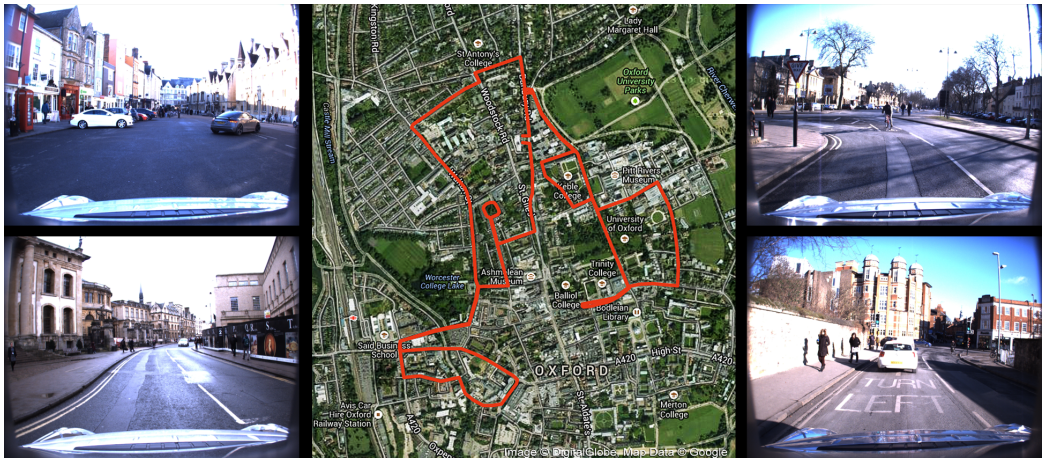


Figure 3.3: Satellite map of the Oxford 10km road dataset with the route indicated in red, flanked by example images from the Oxford datasets captured using the Bumblebee3 camera. Notice the large variation in appearance and structure in a single city.

Table 3.2: Sensors used on the LEAF platform

Sensor	Model	Usage
Lidar	2× SICK LMS-151	Push-broom: mapping, localisation
Camera	Bumblebee XB3	Front–facing: visual odometry, localisation, mapping, perception
	2× Grasshopper2	Left and right facing: mapping, perception, localisation

3.2.1 Oxford 10km road dataset

Oxford's own RobotCar has been on the road recording its excursions since 2012. More recently, there has been a concerted effort to log 1000km over the course of one year, on a single route around the city. The purpose is to capture the natural course of weather, traffic, pedestrians and roadworks of a single 10km route and release it to the public for the benefit of the robotics community [47].

The varying conditions provide a great opportunity to test the robustness of calibration methods. It is important to mitigate the effects that moving vehicles or shadows may have on the calibration. A large dataset like this allows us to study scene-induced calibration in depth, and provide means to ensure we avoid anomalies in the data, which we cover in detail in Chapter 6. Example scenes from these datasets are shown in Figure 3.3.

3.3 GEM Survey Vehicle

The GEM is a small electric vehicle and is equipped with an array of mono- and stereo-cameras, planar and 3D laser scanners, inertial sensing and GPS. It was designed for the purpose of generating detailed surveys of the walkways of Milton Keynes as part of an autonomous transport project. It generates about one terabyte of raw data per hour so there is no shortage of it for research purposes. The information from each sensor can be complementary to that of another. For example, a reliable 3D

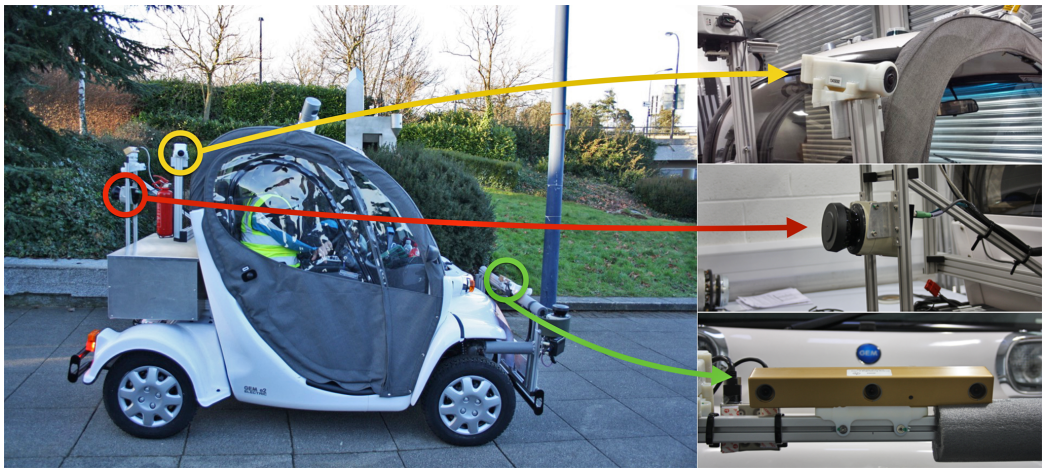


Figure 3.4: The sensors to be calibrated mounted on the platform such that they do not have overlapping FoV. We use the stereo camera mounted at the front of the vehicle to estimate vehicle pose using VO. The data used for calibration was collected in Milton Keynes, UK.

detection system [86] can be used to augment the learning of a computer vision detection system. The viable fusion of sensors is contingent on the accurate cross-calibration of applicable sensors. We therefore focus on ensuring that the cameras and laser scanners are adequately calibrated to perform such tasks.

Table 3.3: Sensors used on the GEM platform

Sensor	**Model**	**Usage**
Lidar	SICK LMS-151	Push-broom: mapping, localisation
	SICK LMS-151	Horizontal: perception, mapping, localisation
Camera	Bumblebee2	Rear-facing: visual odometry, localisation, mapping
	Bumblebee XB3	Front-facing: visual odometry, localisation, mapping
	2× Grasshopper2	Left and right facing: mapping, perception, localisation

The GEM, shown in Figure 3.4, has a PointGrey Research (PGR) Bumblebee XB3 stereo camera mounted front-facing, and tilted downward by approximately 18°. A Bumblebee2 (not indicated) is also mounted at the rear facing backwards. A SICK LMS-151 2D lidar is positioned at the back of the vehicle and is tilted back by approximately 9° as a 'push-broom'. Two PGR Grasshopper monocular cameras are



Figure 3.5: Example camera images from Milton Keynes dataset. The first column show two images from the front-facing stereo camera while the second and third column show images taken from wide-angle left- and right-facing cameras respectively.

mounted at the sides, one facing right and left (not shown) respectively.

3.3.1 Milton Keynes dataset

The LUTZ autonomous transport project at Milton Keynes demonstrated the capabilities of ORI and Oxbotica's collaboration. The project relied on detailed 3D maps of individual routes recorded over a period of a year. These maps created over multiple seasons and weather condition attempts to represent the varying appearance of the natural world, as with the Oxford 10km datasets.

The landscape of the route is predominantly urban, although the GEM is driven mainly on walkways rather than a road occupied by other vehicles. Example scenes from these dataset, captured by the cameras are presented in Figure 3.5. The dataset covers approximately 4km.

The Milton Keynes dataset offers a 'best-case' scenario for real-world calibration. The routes are driven at a walking pace of 3–5km/h allowing a more dense laser swathe to be generated while mitigating likelihood of camera blur in low-light scenarios.

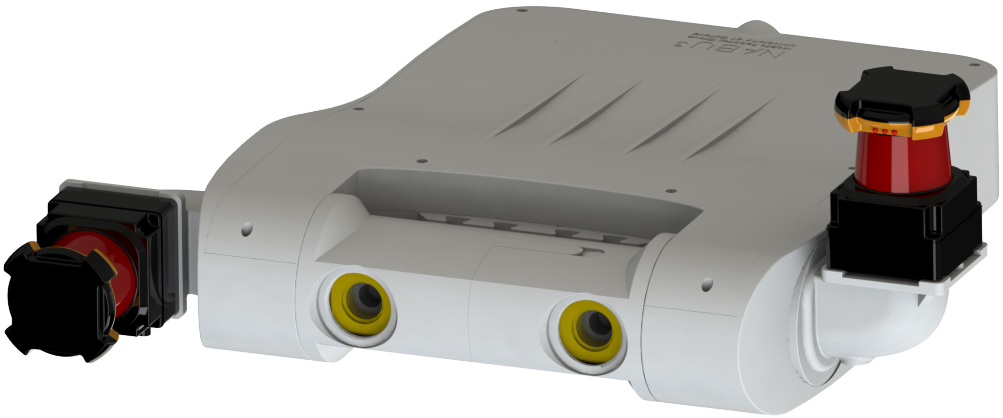


Figure 3.6: Computer rendering of *NABU 3* with two laser scanners configured in 'push-broom' for SLAM, and horizontally for object detection and scan matching.



Figure 3.7: Warehouse used for the *NABU 3* dataset

3.4 *NABU* surveying device

The *NABU*[1] is a device that is designed to be the portable embodiment of the ORI's autonomous stack. The bare-bones system consisting of a stereo-camera, laser scanners and INS is designed to go everywhere. We do not cover this system in detail here but refer the reader to Chapter 2 which details the design, construction and testing of the next generation device as part of the contributions of this thesis. We present a list of the relevant sensors in Table 3.4, which conveys the device rendered in Figure 3.6.

[1] *NABU*, formerly 'Naboo' named after the planet from Star Wars, was renamed to a suitably contrived acronym: NAvigation Base Unit.

Table 3.4: Sensors used on the *NABU 3* platform

Sensor	**Model**	**Usage**
Lidar	2× Hokuyo UTM-30LX	Push-broom: mapping, localisation
Camera	Bumblebee2	Visual odometry, localisation, mapping
	2× Grasshopper2	Left and right facing: mapping, perception, localisation

3.4.1 Warehouse dataset

Calibration for the *NABU 3* was tested using a recorded traversal of a warehouse, pictured in Figure 3.7. The total traversal length was 690m over 20 minutes. The traversal speed was greater than 0.8m/s for 27% of route. This dataset is particularly challenging due its relatively short length and lack of planar structure and features.

Chapter 4

Background

In this chapter we review the literature (Section 4.1) and how each contribution proposes to solve the data association problem and the subsequent pose estimation problem. We discover that certain approaches do not require explicit data association, which is useful for the purpose of 'automatic' calibration.

We take care to note when certain approaches are promoted as 'automatic', rather than 'manual'. Calibration is a composition of many sub-processes, each of which can require more or less human intervention. In the literature, the categorisation of 'automatic' is often given only to one or few of these sub-processes.

Latter sections introduce the relevant concepts of pose geometry and projections. Section 4.5 lays the theoretical groundwork for Normalised Information Distance (NID). This statistical tool measures the mutual dependency between two variables, typically camera grayscale value and laser return intensity when applied to camera–laser calibration.

Section 4.6 covers optimisation techniques ubiquitous in the field of simultaneous localisation and mapping (SLAM), and the challenge of solving for 3D pose. The correct method for a given application depends on the smoothness and convexity of the given cost function, availability of gradients and dimensionality, among other

factors. We discuss how different methods are suited to our optimisation problem. We find that the cost function can often be non-smooth, leading us away from gradient-based methods.

4.1 Related Work

The calibration methods presented in the literature can be broadly divided into the two categories of *manual* and *automatic*. Early methods relied on manual processes of generating and placing calibration targets and selecting features in the laser data [84, 93] or point correspondences in the laser and camera data [70, 95].

Methods later emerged that automated or improved on the sub-processes, such as the automatic line extraction used in [36] to improve on the method in [95]. More recently, assisted by greater availability of real-world data, researchers have developed methods of laser–camera calibration that forgo the checkerboards or fiducial markers; a necessary step towards true autonomy.

4.1.1 Calibration targets

The somewhat recent surge in the popularity of lidar and camera fusion gives rise to a new challenge; the relative position of these sensors needs to be accurately known. Wasielewski and Strauss [90] designed a calibration target consisting of two planes intersecting at an edge. This edge is distinctly observable in both the camera image and laser data and therefore can be strongly associated.

Computer vision tools have improved drastically since 1995, and are useful for the task of extrinsic sensor calibration. On the forefront of this development were Zhang and Pless [93], whose work hinges on the well established field of camera geometry [31] and pose recovery [94].

The rigid transform between the camera and the planer laser scanner is estimated by introducing a coplanar constraint between the laser points and the observed

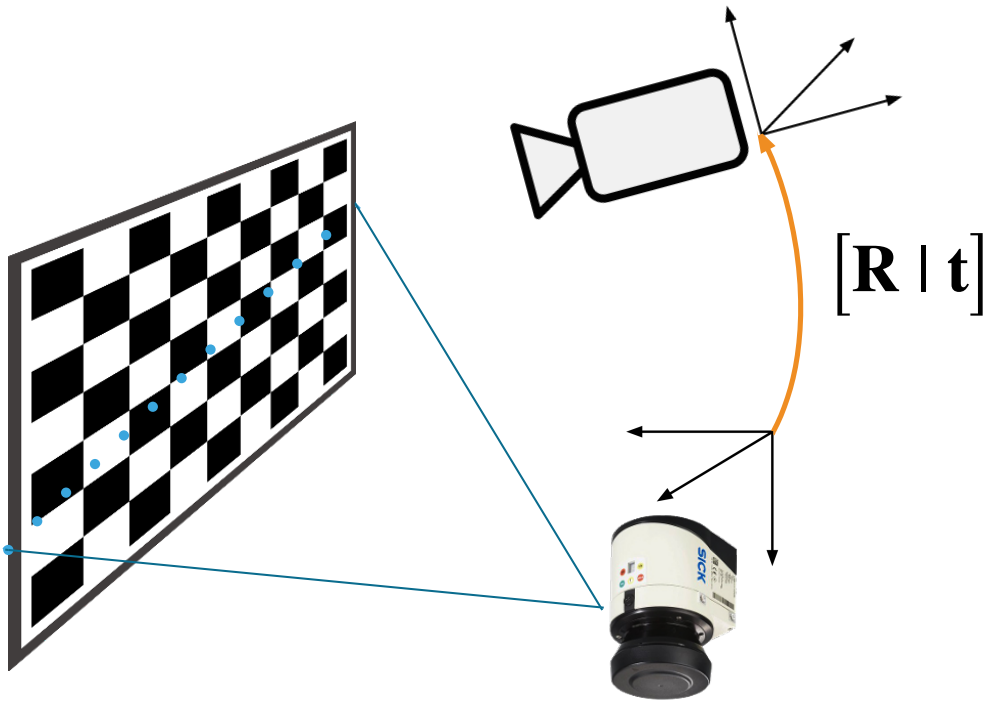


Figure 4.1: This schematic shows the setup of the calibration problem in [93] and others. The checkerboard plane is simultaneously observed and the 3D laser points are constrained to lie on the plane discovered by the camera. $[\mathbf{R} \mid \mathbf{t}]$ represents the rotation and translation of the rigid transform.

checkerboard, typically taking the form

$$\mathbf{n}(\mathbf{R}\mathbf{x} + \mathbf{t}) = \mathbf{0}.$$

The normal of the checkerboard plane with respect to the camera is given as $\mathbf{n}$ and $[\mathbf{R}|\mathbf{t}] \in \mathfrak{se}(3)$ is the transform between the two sensors. This particular setup is shown in Figure 4.1.

The pose of the camera relative to the checkerboard is estimated using a calibration toolbox such as [94]. The laser data points that correspond to the checkerboard are then segmented by hand, and the camera–laser pose can be found by minimising the distance between these points and the checkerboard plane. Maintaining the principle of point-to-plane constraints, this approach has been extended for use with 3D laser scanners [84], multi-planar scanners [33] and omni-directional cameras [60].

Li et al [42] propose a different calibration target, a right-angled triangle, that has distinct edges. These edges can be extracted as line features in the image and as points found at discontinuities in the laser scan. The optimisation seeks to minimise the sum of distances between the points and corresponding lines in image space.

In a similar vein, the authors of [53] associate a detected edge in the camera image with the corresponding laser point. They formulate a closed-form solution for the relative pose

The approach offered by [2] also differs in that they use a single large fiducial marker for plane and camera pose estimation, and an iterative closest point algorithm to minimise point-to-plane error.

These calibration problems are similar in that they require altering the scene to provide the sensors with known geometry to constrain the optimisation. The subprocesses of detecting checkerboard corners, planes and associations between sensors are offered with a varying degree of autonomy; the framework presented in [2] claims an automatic capture and optimisation process eschewing the need for an initial seed.

4.1.2 Correspondence selection in natural scenes

The previous subsection discussed calibration by providing known geometries and allowing the algorithm to extract features and perform data association. Here we discuss calibration methods that do the inverse; observe unaltered natural scenes and whereby a user provides geometric feature matches.

Scaramuzza et al [70] rely on manually selecting point correspondences in the laser and camera data. They aid the user by generating a synthetic laser image, coloured according to the angle subtending the laser scanner and two consecutive laser points, or *bearing angle*. The effect is to make features and edges appear more distinct.

The relative pose can be found using the well known algorithm *Pose from n-Points* (PnP) [66]. The authors then use a non-linear refinement step as the result from the closed-form PnP can be imprecise. An alternative method of calibrating using point

correspondences is offered by Zhao et al [95], which uses direct linear transformation to solve for the camera–laser pose.

Gong et al [28] present a method that exploits an urban scene's common geometric structure. A trihedron, formed by the corner of a wall, can be segmented in the laser and camera data, and constrains the problem sufficiently given three roughly orthogonal planes. Again, segmenting the planes in both sensors' observations is a manual operation.

The only feature-matching solution we review that automatically associates features between the two modalities is by Moghadam et al [51]. The authors propose an algorithm that associates line features detected in the image using the Hough transform, to lines discovered in the laser data at intersecting planes and at plane boundaries.

4.1.3 Global registration in natural scenes

The introduction of the Wildcat, an off-road predecessor to the LEAF in Section 3.2, allowed for large-scale surveying of roads. Napier and Newman [52] take advantage of this by developing a data-driven method of calibration that seeks to align edges in a synthetic laser image with camera image of the same scene.

We cover exactly how this synthetic laser image is generated in Section 4.3. Generating the laser edge image requires convolving the original laser image with derivative kernels such as the Sobel filter

$$p_u = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}, p_v = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}.$$

Both the synthetic and camera edge images are then Gaussian-smoothed and patch-normalised. The post-processed data is visualised in Figure 4.2 where both edge images are then directly compared using the sum of the squared difference between



Camera *Laser*

Figure 4.2: The top row shows the data as read from the sensor while the bottom row shows the corresponding normalised gradient image generated using [52]. Scene taken from the Milton Keynes dataset.

the two. If $\mathcal{I}_k^l({}^cT_l)$ denotes a synthetic laser image patch, and $\mathcal{I}_k^c$ a grayscale camera image patch, each smoothed by Gaussian kernel $Q(\mathbf{x})$, then the objective function is given as

$$O({}^cT_l) = \sum_{\mathcal{I}_k^c} w_{\mathcal{I}_k^c} \left\| Q(\mathcal{I}_k^c) - Q(\mathcal{I}_k^l({}^cT_l)) \right\|^2,$$

where ${}^cT_l \equiv G(\mathbf{g}_{CV} \oplus \mathbf{g}_{VL})$ using our more general notation.

The authors note that the performance of the calibration algorithm depends largely on the appearance of the scene; certain scenes cause the basin of convergence to become non-smooth and non-convex. They demonstrate improved calibration accuracy by combining solutions from multiple scenes using a recursive Bayes filter.

We can contrast the approach of Napier and Newman to that of Levinson and Thrun [41], who propose a method of registering edge features of the camera image with spatial discontinuities in the laser data (as opposed to reflectance discontinuity).

In the case of [41], discontinuity of laser data is given as a function of euclidean distance between adjacent laser points. To enlarge the basin of convergence, the authors create an inverse distance transform of the derivative camera image, effectively estimating the distance of a given pixel to the nearest edge. The two edge modalities, sharing a location in image space, are correlated by taking their product.

The authors show that the algorithm can calibrate a system given even harsh lighting and scene conditions, and that it can track gradually shifting sensors.

Perhaps significantly, the laser used is a Velodyne HDL-64E which would capture geometric discontinuities from a similar viewpoint to that of the camera. This is in contrast the approach of Napier and Newman who use a 2D laser scanner, and we are therefore reserved in comparing these two approaches on direct terms.

4.1.4 Registration using information metrics

To be covered in Section 4.5, Mutual Information is an information theoretic measure of the co-dependence between two random variables. It has been used extensively and effectively for the registration of medical images captured using different technologies [65], for example magnetic resonance (MRI) and computed tomography (CT) imaging.

Mutual information has been used for visual servoing [20], and shows robustness to changes in lighting conditions where a direct approach may introduce non-smoothness and non-convexity in the cost function. The authors report an advantage of using mutual information being the accurate registration without the need for feature extraction and matching.

Mastin et al [49] show successful registration between an aerial image (luminance) of an urban scene and a 3D scan of the same scene encoded with altitude and what they term *probability of detection* (pdet), which is equivalent to the term *reflectance*

we use in this thesis. An interesting outcome is the success of registration using terrain elevation and image luminance, two (seemingly) statistically independent measurements. The authors use an approximation of the joint entropy of these two variables coupled with the luminance from the camera image,

$$H(u, v_e, v_p) = H(u, v_e) + H(u, v_p)$$

where u, v_e, v_p are the image luminance, lidar range (elevation) and pdet values respectively. Their results show effective registration and also indicate that using both elevation and lidar intensity in the evaluation jointly gives an improvement in registration performance over using either independently.

There are a number of publications that deal specifically with terrestrial mobile robots, and register camera and laser data using mutual information. Taylor and Nieto [81, 82] perform registration of a high-fidelity laser scanner and hyperspectral camera. Their formulation of the cost function applies a *normalisation* that mitigates the effect that the amount of data 'overlap' has on the total value. They choose a *particle swarm* [24] optimiser to solve for the extrinsic parameters due to the cost function being noisy and having multiple local optima.

Pandey et al [62] and Wang et al [88] employ lower fidelity sensors and show successful calibration by registering laser reflectance with camera luminance using mutual information. Their implementation registers a camera with a wide-angle lens to a 360° Velodyne scanner.

4.2 Operations and conventions in SE(3)

Calibration is formulated as an optimisation over the relative pose between two frames of reference, thus it is important to choose an appropriate parameterisation of this pose. We provide here an overview of the parameterisations used in this thesis and refer the reader to the literature for a more in-depth review [13].

The set of all possible rotations and translations in 3D space lies on the manifold of the Special Euclidean group $\mathbf{SE}(3)$. This set can be minimally represented using six parameters. Translation is represented by the vector $\mathbf{t} \in \mathbb{R}^3$, while rotation can be parameterised using Euler angles ($\boldsymbol{\theta} \in \mathbb{R}^3$), axis-angle ($\boldsymbol{\phi} \in \mathbb{R}^3$) or quaternions ($\mathbf{q} \in \mathbb{R}^4$).

The **Euler angle** representation is minimal and most intuitive, and can be expressed as *roll*, *pitch* and *yaw*; rotation about the respective orthogonal x, y and z axes. However, it suffers from a singularity known as 'Gimbal lock', a phenomenon that, depending on order of representation, occurs at a pitch angle of $+\frac{\pi}{2}$ or $-\frac{\pi}{2}$.

Quaternions over-parameterise rotation, thus avoiding the singularities that minimal representations may encounter. Optimisation using quaternions requires that the vector element of the quaternion be constrained to unit length, in addition to maintaining an extra parameter.

The $\mathbf{SO}(3)$ group is a **Lie group**, and thus has an associated **Lie algebra** $\mathfrak{so}(3) \in \mathbb{R}^3$, which is conveniently a minimal parameterisation of rotation. The Lie algebra is transformed to its Lie group using the exponential map $\mathbf{R}(\cdot) \ : \ \mathfrak{so}(3) \mapsto \mathbf{SO}(3)$, and is represented by the skew symmetric matrix given by the operator $\lfloor \cdot \rfloor_\times$ as

$$\lfloor \boldsymbol{\phi} \rfloor_\times = \begin{bmatrix} \phi_x \\ \phi_y \\ \phi_z \end{bmatrix}_\times = \begin{bmatrix} 0 & -\phi_z & \phi_y \\ \phi_z & 0 & -\phi_x \\ -\phi_y & \phi_x & 0 \end{bmatrix} \tag{4.1}$$

The relationship between the Lie algebra and its corresponding Lie group can be expressed by the exponential and logarithmic maps. The map $\exp : \mathfrak{so}(3) \mapsto \mathbf{SO}(3)$, which forms the rotation matrix is given as $\mathbf{R}(\boldsymbol{\phi}) = e^{\lfloor \boldsymbol{\phi} \rfloor_\times}$.

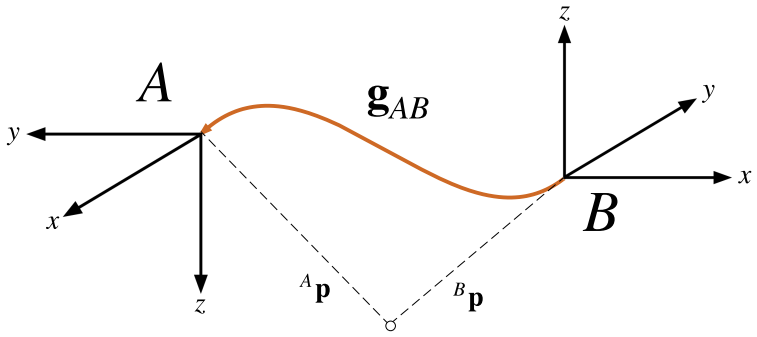


Figure 4.3: Notational conventions for transforms in **SE**(3). The relative transform *from* frame B *to* frame A is expressed as $\mathbf{g}_{AB}$. Thus, to express the point ${}^{B}\mathbf{p}$ with respect to frame A requires : ${}^{A}\mathbf{p} = \mathbf{g}_{AB} \oplus {}^{B}\mathbf{p}$.

4.2.1 Notation

Here we define the notation used in this thesis to express relative transforms of **SE**(3) between two frames of reference. We use the Lie algebra representation to formulate the tuple $\mathbf{g} = (\boldsymbol{t}, \boldsymbol{\phi}) \in \mathfrak{se}(3) \subset \mathbb{R}^6$. As shown in Figure 4.3, $\mathbf{g}_{AB}$ denotes the transform *from* frame B *to* frame A, also indicated by the direction of the arrow. A point represented in frame B can be expressed in frame A using ${}^{A}\mathbf{p} = \mathbf{g}_{AB} \oplus {}^{B}\mathbf{p}$.

Throughout this document we will use the composition operators $(\oplus, \ominus)$ to denote transforms between $\mathfrak{se}(3)$ elements.

4.3 Point cloud swathe generation

Calibration requires the association of features in the scene co-observed by both sensors. These features could be points, lines, planes or edges which are represented in either a common world $\mathbb{R}^3$ or image $\mathbb{R}^2$ space. This section covers how we model the sensors such that features can be represented in either space, whereby the latter simply requires a camera projection of the former.

3D point clouds are often generated using a Velodyne 360° laser scanner, which necessitates that a camera's field of view will overlap with the scanning field. We cover how we represent features in a common space when the sensors' fields of view

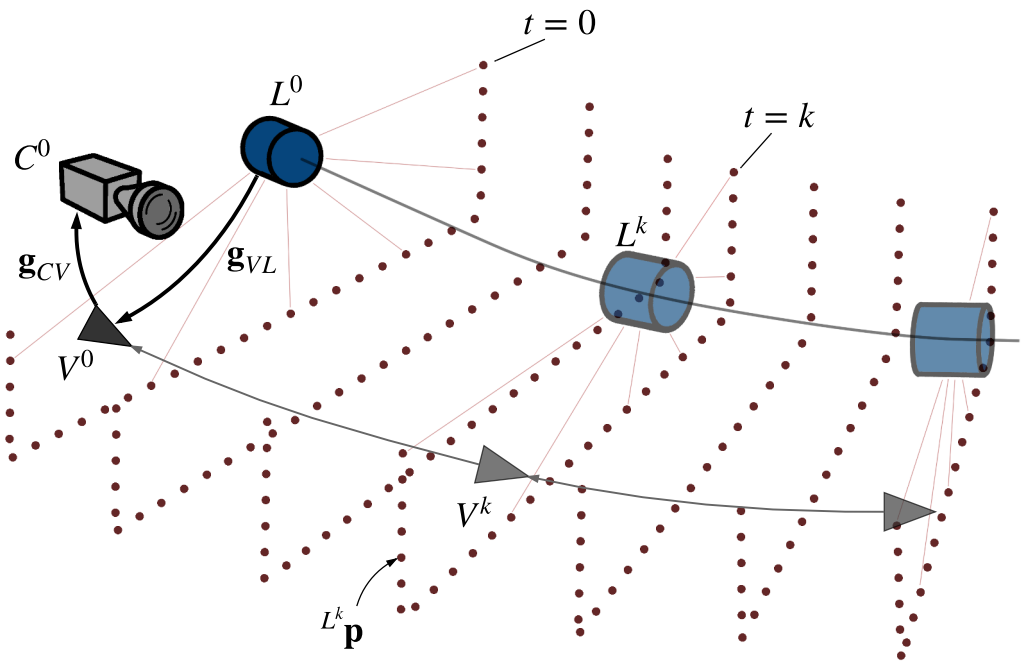


Figure 4.4: This schematic illustrates how a 3D point cloud *swathe* is generated using known ego-motion and 2D laser scanner. Consider the camera C, laser L and vehicle V frames as rigidly connected. The ego-motion transform between times 0 and k is represented by $\mathbf{g}_{V^0V^k}$, while a point in the laser frame at time k is represented by ${}^{L^k}\mathbf{p}$. The resulting swathe can thus be projected as required into the camera frame C^0.

do not overlap, a common occurrence in robot setups. We rely on the ego-motion to obtain an *eventual* overlap in observations, eschewing the need for simultaneous observations (assuming a static environment). The configuration of the 2D laser scanner, as described in Section 3.3, is such that we can generate a 3D point cloud simply by driving forward. We refer to this configuration as 'push-broom'. Knowing the precise time at which the laser measurement occurred, and its corresponding location along a known ego-motion, the position of that point relative to a reference frame along the ego-motion can be determined. Our calibration uses the ego-motion from the visual odometry of a stereo camera, but other locally accurate sensors such an inertial navigation system, or even horizontal scan-matching [95] can be used.

A common issue when dealing with measurements of multiple sensors is that of clock synchronisation. We have to be sure of the time a particular laser scan has

been taken relative to a camera image. For this we use the time synchronisation tool, TICSync [30].

In reference to Figure 4.4, our intention is to express a laser point, ${}^{L^k}\mathbf{p}$ measured at time k, in the coordinate frame of a camera at time 0. This requires accounting for ego motion between times 0 and k, which is given as $\mathbf{g}_{V^0V^k}$. The coordinate frames for *vehicle*, *camera*, and *laser scanner*, are denoted V, C and L respectively. The point can be thus expressed as

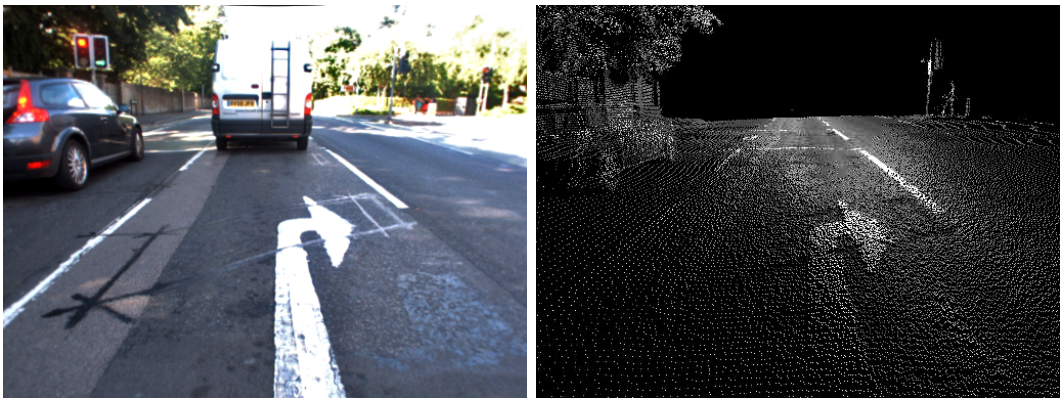
$$ {}^{C^0}\mathbf{p} = \mathbf{g}_{CV} \oplus \mathbf{g}_{V^0V^k} \oplus \mathbf{g}_{VL} \oplus {}^{L^k}\mathbf{p}, \tag{4.2} $$

where $\mathbf{g}_{V^kL^k}$ is the rigid transform from the laser frame to the vehicle (ego-motion) frame. The rigid camera–vehicle–laser transforms remain constant between robot poses, thus $\mathbf{g}_{VC} \equiv \mathbf{g}_{V^tC^t}$ and $\mathbf{g}_{VL} \equiv \mathbf{g}_{V^tL^t}$ for all t, therefore we drop the time notations in these instances. In the case of a stationary vehicle, Equation (4.2) simplifies since the vehicle frame becomes redundant, leaving

$$ {}^{C^0}\mathbf{p} = \mathbf{g}_{CL} \oplus {}^{L^0}\mathbf{p}. \tag{4.3} $$

4.4 Camera and projection models

The final requirement for calibration is that the laser points are projected into the relevant camera frame. A point is projected from $\mathbb{R}^3$ into the $\mathbb{R}^2$ space of the camera (at time $t = 0$) using the calibrated camera model parameters. For the purpose of this thesis we always assume that intrinsic camera calibration is available, thus presuming the focal lengths $\{f_x, f_y\}$ and principal point offsets $\{c_x, c_y\}$ to be known. We therefore use the following projection model



a. b.

Figure 4.5: Elements of the calibration scene: The camera image (left) alongside a synthetic laser image created using $\{\mathbf{x}, {}^{C}\mathcal{I}\}$

$$
\begin{aligned}
{}^{j}\mathbf{x}_k &= \Pi\left({}^{C_j}\mathbf{p}\right) \\
\text{where} \quad \Pi(\cdot) &= \begin{bmatrix} f_x \frac{x}{z} + c_x \\ f_y \frac{y}{z} + c_y \end{bmatrix}
\end{aligned}
\tag{4.4}
$$

We can denote $\mathbf{x} \subset \mathbb{R}^2$ as the set of all points in image space at time $t = 0$ for all laser points ${}^{L^k}\mathbf{p} : 0 \leq k \leq \tau$, where τ is the designated end–timestamp of the swathe. Since the points are represented in a space the same as the image, it is possible to measure similarity of scene as observed from the camera and the laser, or to determine the reprojection error between point or line features.

4.4.1 Creating a synthetic image of a scene

Some of the calibration methods we discuss in Chapter 5 take advantage of the measured intensity of laser return. Each laser point ${}^{L^k}\mathbf{p}_i$ will have an associated intensity value ${}^{L}\mathcal{I}_i$. The set of spatial points and intensity $\{\mathbf{x}, {}^{L}\mathcal{I}\}$ allows for the creation of a synthetic image of the scene, of which Figure 4.5 shows an example.

The first problem the reader may notice is that the image is sparse, with points concentrating in the distance. As-is, the image cannot be processed using spatial

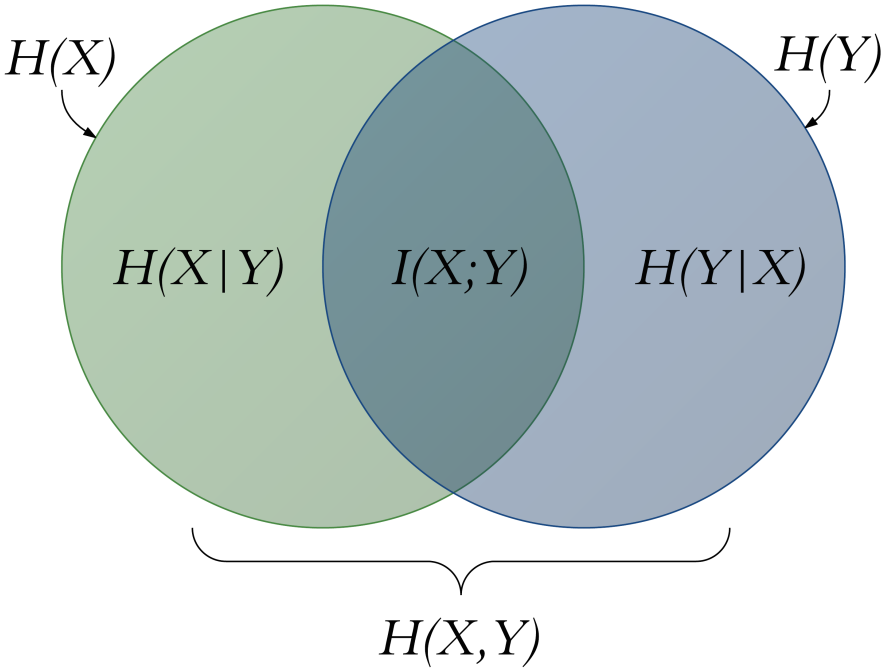


Figure 4.6: Venn diagram of bi-variate Entropy.

filters such as edge filters or feature detectors. The information theoretic metrics we use in Section 5.1 alleviate this problem, yet we also have the option to recover the missing data by interpolation since the resulting image is sparse.

4.5 Information Theory

Information theory is the study of how information is encoded in a sequence of data. Specific to our problem of calibration, we would like to measure the statistical dependency of an observation from one sensor on another. This section provides the reader a brief introduction to the information theoretic measures used to formulate the calibration objective function.

4.5.1 Entropy

Entropy is a measure of statistical unpredictability, or 'randomness', and is used to quantify the information — or the uncertainty thereof — carried in a signal or

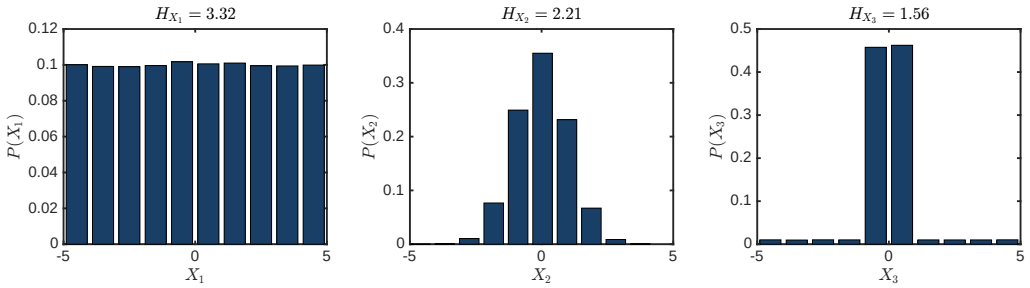


Figure 4.7: Examples of distributions and their entropy. Distributions with higher 'randomness' have higher entropy

distribution. The series of histograms in Figure 4.7 offer a visual aid, showing how a greater certainty leads to lower entropy. An entropy of zero indicates the variable is deterministic. The expression for entropy is given formally as the following integral for a continuous probability mass function,

$$H(X) = -\int \mathbf{P}(x)\log(\mathbf{P}(x))dx.$$

The formulation of entropy for a discrete histogram, analogous of a continuous distribution, is given as

$$H(X) = -\sum_{i=1}^{n} \mathbf{p}(x_i)\log(\mathbf{p}(x_i)). \tag{4.5}$$

Distinct from the entropy of a continuous distribution, the entropy of a discrete distribution in Equation (4.5) does not depend on the order. This point is clarified when we apply theories of entropy to calibration. We will be interested in the *joint entropy*; the entropy inherent in a joint distribution of two (or more) variables. The joint entropy of two discrete random variables X and Y is given as

$$H(X,Y) = -\sum_{j=1}^{n_y}\sum_{i=1}^{n_x} \mathbf{p}(x_i,y_j)\log(\mathbf{p}(x_i,y_j)). \tag{4.6}$$

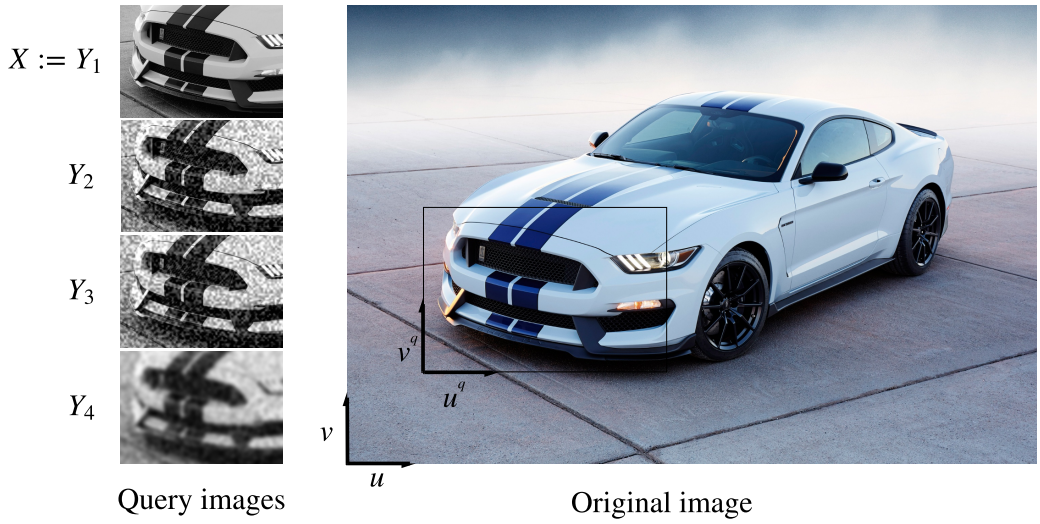


Figure 4.8: Example image registration problem using mutual information and normalised information distance. The query images Y_i are generated, top to bottom, by 1) cropping, 2) adding random noise, 3) add Gaussian blur with $\sigma = 2$ and 4) add Gaussian blur with $\sigma = 6$. The registration problem aligns Y_i with the original image X using mutual information by solving $\arg\max_{u,v} I(X(u,v);Y_i)$.

4.5.2 Mutual Information

Mutual Information measures the mutual statistical dependence between two random variables. It can be expressed analogously using a Venn diagram as shown in Figure 4.6, and is represented by the overlapping region. Intuitively, we are quantifying the information we gain about Y by knowing X, and vice versa. Mutual information for discrete distributions can be expressed formally as

$$I(X;Y) = -\sum_{j=1}^{n_y}\sum_{i=1}^{n_x} \mathbf{p}(x_i,y_j)\log\left(\frac{\mathbf{p}(x_i,y_j)}{\mathbf{p}(x_i)\mathbf{p}(y_j)}\right), \tag{4.7}$$

or as a combination of marginal and joint entropies

$$I(X;Y) = H(X) + H(Y) - H(X,Y). \tag{4.8}$$

4.5.3 Normalisation

Mutual information, given above, has a lower bound of zero but no upper bound. Typically measured in *bits*, Shannon Entropy [73] of two data streams of differing lengths cannot reasonably be compared since the entropy is dependent on this length. Thus, a normalisation that bounds the measure between zero and one is desirable.

Normalising accounts for the length of a data stream, creating a relative measure of information contained in a data stream. Normalisation is shown to be possible by observing that $I_{X;Y} \leq \min(H_X, H_Y)$. *Normalised Mutual Information* [79] can be defined using the geometric mean of H_X and H_Y as follows

$$NMI(X,Y) = \frac{I(X;Y)}{\sqrt{H(X)H(Y)}}. \tag{4.9}$$

Notice that for $I_{X;X} = H_X$, the upper bound $NMI_{XX} = 1$, is achieved.

Normalised information distance [43] uses the joint entropy as a normalisation factor, and is defined

$$NID(X,Y) = 1 - \frac{I(X;Y)}{H(X,Y)}. \tag{4.10}$$

The distance is again bounded between zero and one, albeit where maximal similarity is indicated at the lower bound. Again, we can prove the lower bound using the identities $H_{XX} = H_X$ and $I_{X;X} = H_X$,

$$NID(X;X) = 1 - \frac{H(X)}{H(X)} = 0.$$

Unlike mutual information, NID is a true metric since it is non-negative, symmetric and satisfies the triangle inequality. We will show later that this makes it convenient and meaningful to sum the distances obtained from multiple camera–laser observations.

To explain the difference between mutual information and joint entropy, we use an

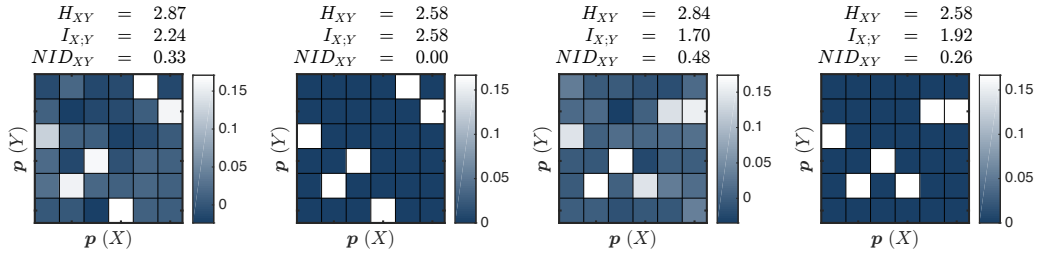
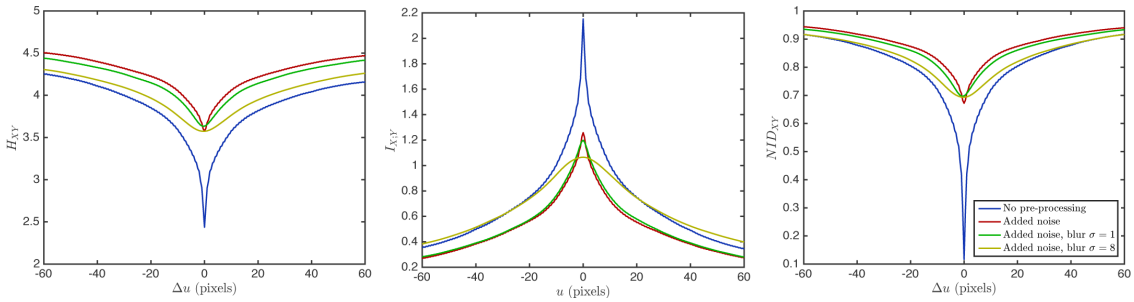


Figure 4.9: How changing the joint distribution (histogram) of two variables affects Entropy, Mutual Information, and Normalised Information Distance. The first two histograms (one affected by simulated noise) represent an 'ideal' distribution where each $P(X)$ has a unique $P(Y)$ and vice-versa. The latter two do not exemplify this. As a result, the mutual information is lower (NID is higher), yet joint entropy remains unchanged.



a. b. c.

Figure 4.10: Comparing the cost surfaces of (a) Joint Entropy, (b) Mutual Information and (c) Normalised Information Distance under varying effects of noise and blur.

illustrative example in Figure 4.9. Presented are four example joint histograms. The former two exemplify a distribution which offers a unique value of X for a given Y (one with simulated measurement noise), and vice versa. In the noiseless, ideal case, the NID is zero; information between X and Y is shared perfectly. The latter two do not have unique value of X for a given Y, thus information is not perfectly shared. This is reflected in the values of MI and NID, which we expect to be higher and lower respectively than the first two distributions. Joint entropy has however remained unchanged, thus making it unfavourable as a measure of mutual dependency.

We demonstrate how image registration, or *visual servoing*, makes use of information metrics. Given an image, we use the aforementioned measures to correlate a

query patch Y_i with the underlying image X. In our example we corrupt the query patch by applying blur and noise, as shown in Figure 4.8. By perturbing the pixel location of the query patch Y_i from the correct position we plot the cost surface of each of the information metrics in Figure 4.10. We intend to demonstrate two phenomena; the archetypal shape of cost surface around the solution, and the widening and smoothing of the cost surface due to blur despite added random noise.

We learn from the example that objective functions of image registration using measures of entropy are non-convex, but would be classified as *quasiconvex* since they are monotonic functions. The mathematical definition for quasiconvexity of a function over the subset of real numbers S is

$$f(\alpha x_1 + (1-\alpha)x_2) \leq \max(f(x_1), f(x_2)) \;\; : \;\; \forall x_1 \leq x_2, \quad x_1, x_2 \in S.$$

This definition will become more important in Chapter 6 which characterises the shape of the objective function for the purposes of improving calibration.

4.6 Nonlinear Optimisation

We ultimately want to estimate the relative transform between the two sensors. Invariably this requires solving a nonlinear problem due to the effects of rotation — not least the nonlinearities invoked by the NID operation. Thus we form of the general optimisation as a minisation over the parameters we seek

$$\mathbf{g}^* = \arg\min_{\mathbf{g}} \;\; f(\mathbf{g}). \tag{4.11}$$

Commonly, calibration (or localisation) based on the distance of geometric features is obtained using gradient descent method of Levenberg–Marquardt [60, 70, 93], a variant of the Gauss–Newton method. In this section we discuss methods of optimisation and suitability of each to our particular calibration problem, and how gradient

descent is not always most suitable. Presented here is an brief summary of optimisation strategies relevant to this thesis. This is not intended to be a comprehensive resource and we refer the reader to the literature on optimisation for a more detailed review [56].

4.6.1 Gauss–Newton nonlinear least squares

The Gauss–Newton algorithm is a minimiser typically used to solve problems formulated as non-linear least squares of the form

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{r}(\mathbf{x})^\top \mathbf{r}(\mathbf{x}), \tag{4.12}$$

where $\mathbf{r}(\mathbf{x})$ is a vector of residual distances that are non-linear with respect to $\mathbf{x}$. Pose estimation is an apt situation to employ a non-linear least squares optimiser, provided that $\frac{\mathrm{d}\mathbf{r}}{\mathrm{d}\mathbf{x}}$ is readily available. The residual r_i is often the difference between observed or projected landmarks with sometimes thousands of observations. This makes it a suitable choice for minimising the reprojection error of point features, particular to our calibration methods in Section 5.2 and Chapter 7.

Considering the update equation of Newton's method for the scalar case,

$$\mathbf{x}_{\alpha+1} = f(\mathbf{x}_\alpha) + \frac{\nabla f(\mathbf{x}_\alpha)}{\nabla^2 f(\mathbf{x}_\alpha)} \tag{4.13}$$

we see that it requires explicit computation of the Hessian with computational cost $\mathcal{O}(n^2)$. This becomes intractable for large problems, thus we use the following approximation of the Hessian. Taking the first derivative of f in Equation (4.12) produces,

$$\begin{aligned} \frac{\mathrm{d}f}{\mathrm{d}\mathbf{x}} &= 2\frac{\mathrm{d}\mathbf{r}}{\mathrm{d}\mathbf{x}}^\top \mathbf{r} \\ \mathbf{G} &= 2\mathbf{J}^\top \mathbf{r}. \end{aligned} \tag{4.14}$$

Then, deriving Equation (4.14) we obtain

$$\begin{aligned} \frac{\mathrm{d}^2 f}{\mathrm{d}^2 \mathbf{x}} &= 2\frac{\mathrm{d}\mathbf{r}}{\mathrm{d}\mathbf{x}}^{\top}\frac{\mathrm{d}\mathbf{r}}{\mathrm{d}\mathbf{x}} + 2\frac{\mathrm{d}^2\mathbf{r}}{\mathrm{d}^2\mathbf{x}}\mathbf{r} \\ \mathbf{H} &\approx 2\mathbf{J}^{\top}\mathbf{J}, \end{aligned} \tag{4.15}$$

where the second term is dropped based on the assumptions that the residuals are relatively small and are distributed about a zero mean. We have the components to formulate the update function by revisiting Equation (4.13) to obtain

$$\begin{aligned} \mathbf{x}_{\alpha+1} &= \mathbf{x}_{\alpha} - \Delta\mathbf{x} \\ &= \mathbf{x}_{\alpha} - \mathbf{H}_{\alpha}^{-1}\mathbf{G}_{\alpha}. \end{aligned} \tag{4.16}$$

$\mathbf{G} = 2\mathbf{J}$ and $\mathbf{H} \approx 2\mathbf{J}^{\top}\mathbf{J}$ correspond the Jacobian and approximate Hessian, respectively, of the problem formulated in Equation (4.12). The update step, $\Delta\mathbf{x}$, is subsequently computed as the closed form solution of the equation,

$$\mathbf{J}^{\top}\mathbf{J}\Delta\mathbf{x} = \mathbf{J}^{\top}\mathbf{r}. \tag{4.17}$$

The Levenberg-Marquardt algorithm [48] is a popular adaptation, and adds a damping factor to the Hessian approximation, effectively modulating the size of the update step. The updated Hessian is thus defined

$$\mathbf{H}' = \mathbf{J}^{\top}\mathbf{J} + \lambda\mathbf{I}, \tag{4.18}$$

where λ is a non-zero scalar. Setting $\lambda = 0$ is equivalent to the standard Gauss-Newton algorithm, while increasing values of λ decrease the impact of the Hessian, thus approaching a 'steepest-descent' algorithm. Crucially, the damping factor also handles the scenario of $\mathbf{J}^{\top}\mathbf{J}$ being rank-deficient and therefore singular.

Optimisation problems involve small, iterative steps in a particular direction to seek out an optimum. Depending on the parameterization of pose, this step operation may not be closed. Both rotation matrices and quaternions are not minimal representation of pose, and therefore are not closed under addition or subtraction;

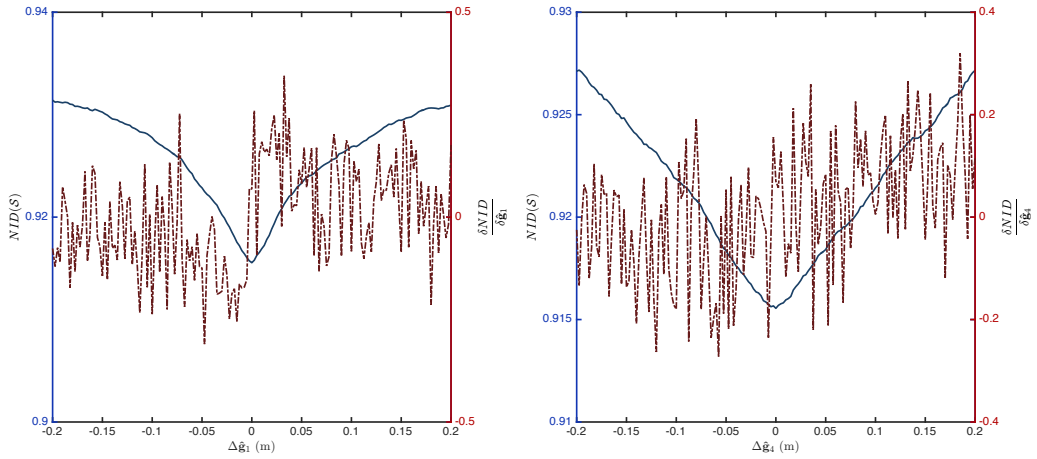


Figure 4.11: NID cost surface and its derivative in the X (left) and *pitch* (right) parameters, with differential $\delta\mathbf{g}$ chosen sufficiently small as to exhibit its non-smoothness. The cost function itself appears smooth to the eye due to the resolution at which the curve is presented.

they are constrained by $\mathbf{R}^\top\mathbf{R} = 1$ and $\mathbf{q}\mathbf{q}^\top = 1$ respectively.

Recall from Section 4.2 that two minimal parameterization exist; Euler angles and the algebra of the Lie group, $\mathbf{SO}(3)$. Since Euler angles suffer from singularities, a common approach is to optimise on-manifold [13, 25] using the $\mathfrak{so}(3)$ tangent space of the Lie group to apply a step. The $\mathfrak{so}(3)$ and $\mathfrak{se}(3)$ Lie algebras are closed under addition and subtraction operations; they remain on the manifold.

4.6.2 Nelder-Mead (simplex)

Gradient-based algorithms may not always be best suited for a particular problem. The calibration algorithm presented in Section 5.1 at times exhibits a non-convex and non-smooth cost function, therefore typical gradient-based methods may either diverge or converge to local minima.

The Nelder-Mead method [54] does not require gradients. Instead, it maintains a set of $n + 1$ data points in $\mathbb{R}^n$, whose convex set forms a simplex. This set can be *expanded*, *contracted* or *reflected* depending on the relative cost evaluated at their location in $\mathbb{R}^n$. The algorithm is presented at the end of the chapter in Algorithm

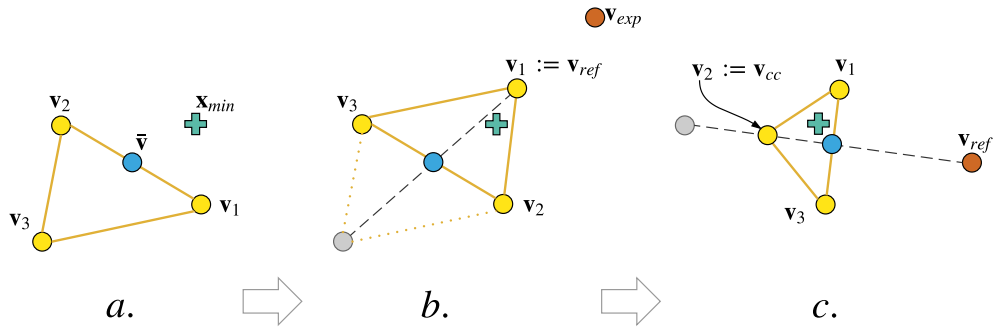


Figure 4.12: A graphical demonstration of two interations of the Nelder-Mead algorithm. Vertices in (a) are ordered according to distance from $\mathbf{x}_{min}$ The point furthest from $\mathbf{x}_{min}$ in (a) is *reflected* about $\bar{\mathbf{v}}$ to find $\mathbf{v}_{ref}$. In (b) $\mathbf{v}_{exp}$ is an *expansion* of $\mathbf{v}_{ref}$, but is further from the solution and is thus rejected. New vertex ordering is established. A *contraction* is performed in (c) where $\mathbf{v}_{ref}$ is rejected.

1, of which two iterations are shown graphically in Figure 4.12. With the aid of [56] we define the notation used in the algorithm. We begin by defining the set of points that form the simplex $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{n+1}\} \subset \mathbb{R}^n$.

At each iteration, a reflection point $\bar{\mathbf{v}}$ is determined by taking the mean of the first n points in $\mathbf{V}$, sorted ascending according to $f(\mathbf{v}_i)$. Explicitly, the vertices are ordered such that

$$f(\mathbf{v}_1) < f(\mathbf{v}_2) < \cdots < f(\mathbf{v}_{n+1})$$

.

The reflection point is calculated as the mean of the n lowest-value vertices, which is given as

$$\bar{\mathbf{v}} = \frac{1}{n} \sum_{i}^{n} \mathbf{v}_i.$$

The operation to be performed (reflection, expansion or contraction) is determined by the relative function value at $\bar{\mathbf{v}}$.

Certainly, the most attractive aspect of the Nelder-Mead method is its relative robustness to non-smooth cost functions. Given the complexity of the cost function

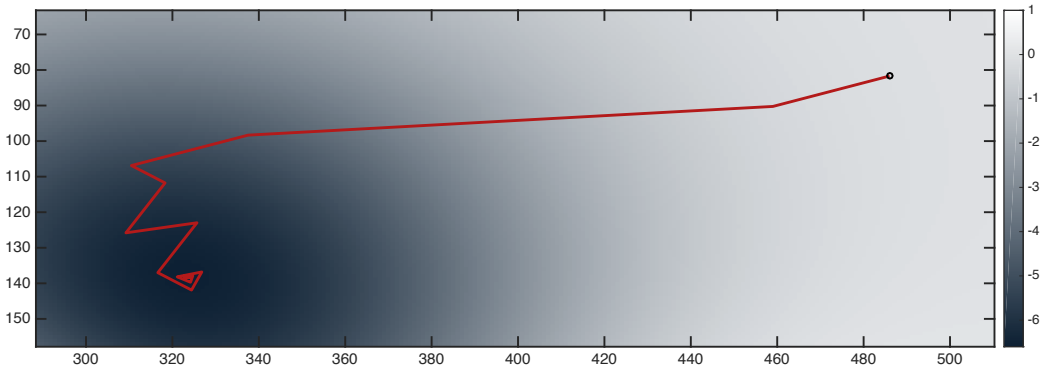


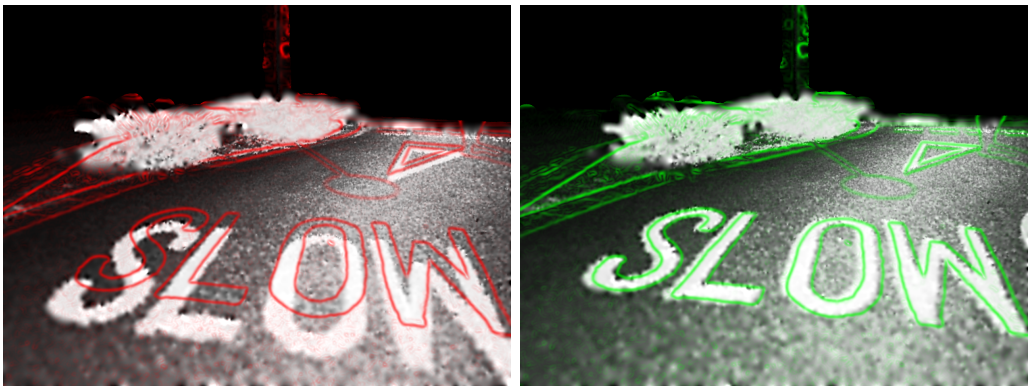
Figure 4.13: Shown here is a cost surface with a minimum located in the dark region. The red trail indicates the zig-zagging propagation of the Nelder-Mead's 'best guess', with the starting point indicated by the black circle.

derived in Section 5.1, analytical derivatives are difficult to formulate and numerical derivatives are susceptible to discontinuities in functions such as histograms. From Figure 4.11, which depicts a typical cost surface and derivative of the NID objective function, it becomes clear that a gradient-based approach would not be suitable.

The computational cost of the Nelder-Mead method is of some concern, since the algorithm has a tendency to 'zig-zag' towards the minimum. This phenomenon is clear in Figure 4.13, and the correct solution is still obtained. Calibration is not a time-critical aspect of robotics, as with localisation. We can therefore afford to sacrifice speed for reliability and accuracy, making the Nelder-Mead method more than adequate for this application.

4.7 Evaluating calibration

The goal of calibration is to accurately measure the relative pose between sensors, and to ensure that this relative pose allows for correct data association between sensor data. There are two modes of analysis; qualitative and quantitative. The former can be done by inspection, ensuring that the fused data is correctly associated, while the latter poses a challenge of its own. This section discusses how we evaluate the calibration methods we propose in this thesis.



Uncalibrated. Calibrated.

Figure 4.14: By visual inspection we assess whether the algorithm has converged to a potential global minimum. The highlighted edges are obtained using the derivative of the original camera image

4.7.1 Measuring calibration performance

Inspection: A human is quite capable of determining whether the discovered data associations are likely correct, thereby providing qualitative verification. Overlaying a transparent *edge* camera image on to the laser data allows for easy recognition of line features; we have visual confirmation of convergence to a local minimum in Figure 4.14 as the edges align closely. Again, we need to tread carefully, since observing correct data associations in 2D (or 3D) may inform us only of a 'good' local minimum.

Ground truth: Quantitative analysis of calibration faces the same obstacle as that of localisation; there is no ground truth. In the case of localisation, new procedures are often compared to reliable pose estimates supplied by high-fidelity GPS and inertial navigation systems.

Calibration solves a single rigid transform between two or more sensors, and demands higher accuracy for reliable operation of higher level subsystems. We therefore require an alternative, reliable method of obtaining these pose values. We instead perform meticulous manual calibration using a checkerboard, with a stationary robot, thus offering the 'best possible calibration'.

We obtain 'ground truth' using a reliable target-based calibration method with a

motion tracker, which we discuss in detail in Section 4.7.3.

Concurrence: While more a measure of *precision*, consistent convergence informs us as to how well constrained the calibration problem is given the parameters and observed scenes. To trust such results as a matter of course would be unwise; any latent bias in the system will invariably go unnoticed. We measure precision according to the variance of the calibration solutions, in a Monte-Carlo sense.

Known calibration: Stereo-camera units have been calibrated to a high precision by the manufacturer. We use this knowledge to compare an inferred stereo-camera calibration to the 'true' one. Consider a well calibrated stereo-camera pair AB with true extrinsic calibration $\mathbf{g}_{AB}$. We calibrate each camera independently with respect to the laser scanner, obtaining two estimates $\hat{\mathbf{g}}_{AL}$ and $\hat{\mathbf{g}}_{BL}$, and thus infer that $\hat{\mathbf{g}}_{AB} := \hat{\mathbf{g}}_{AL} \ominus \hat{\mathbf{g}}_{BL}$. Finally, the error is calculated as $\mathbf{e} = \hat{\mathbf{g}}_{AB} \ominus \mathbf{g}_{AB}$.

To recapitulate, each method of evaluation poses unique problems. We proceed by demonstrating calibration according to a combination of these, supplying corroborating evidence of successful and accurate calibration wherever possible.

4.7.2 Reporting angles

Euler angles offer an intuitive way to report angles but since we are interested in the variance for the sake of evaluation and comparison, it makes sense to use a representation that allows us to do so. We therefore report angles using the axis-angle representation which is not susceptible to singularities in the way Euler-angles are, and are easier to interpret than quaternions.

We report the angles in the form of an average unit vector $\hat{\boldsymbol{\phi}}'$ and a measure of the variance $\boldsymbol{\kappa}$ by modelling them as a von Mises-Fisher distribution. From [6] it is shown that the average unit vector from a set can be obtained by maximising the likelihood, giving

$$\hat{\boldsymbol{\phi}}' = \frac{\boldsymbol{r}}{\|\boldsymbol{r}\|} = \frac{\sum_{i=1}^{n} \boldsymbol{x}_i}{\|\sum_{i=1}^{n} \boldsymbol{x}_i\|}. \tag{4.19}$$

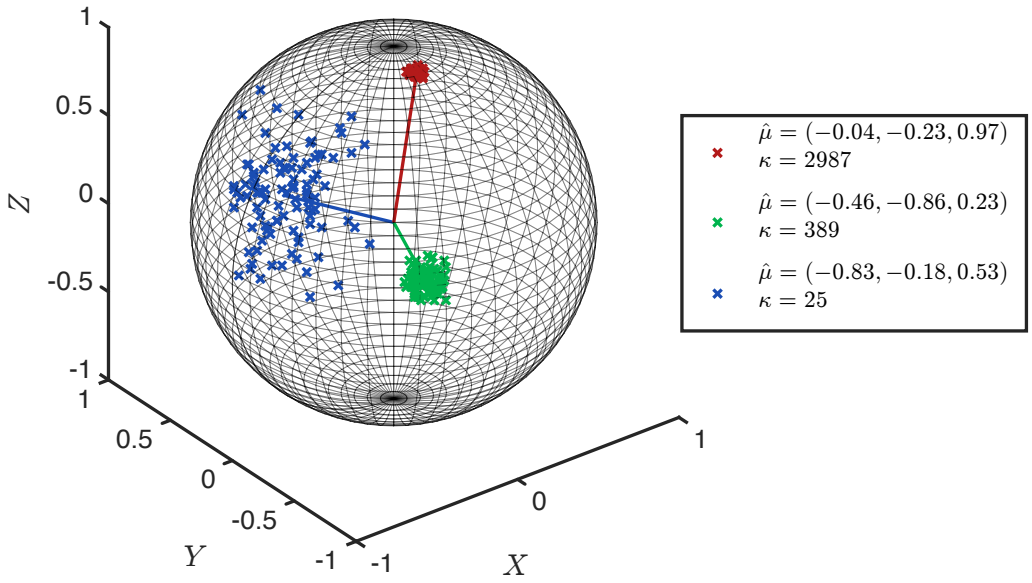


Figure 4.15: A visual guide for understanding von Mises-Fisher distribution. Each cluster represents a different distribution of unit-vectors with mean unit-vector $\hat{\boldsymbol{\mu}}$. Tighter concentrations of vectors on the unit sphere correspond to higher values of $\boldsymbol{\kappa}$.

The concentration parameter $\boldsymbol{\kappa}$ is analogous to variance and measures how the vectors are distributed on the unit sphere. Again, as formulated in [6], the concentration of vectors in $\mathbb{R}^d$ can be estimated by

$$\hat{\boldsymbol{\kappa}} = \frac{\bar{r}d - \bar{r}^3}{1 - \bar{r}^2} \qquad \text{for} \quad \bar{r} = \frac{\|\boldsymbol{r}\|}{n}. \tag{4.20}$$

Unlike variance, the values are unintuitive so we offer Figure 4.15 as visual guide; increasing values of $\boldsymbol{\kappa}$ indicate the vectors are more tightly concentrated.

4.7.3 'Ground truth' calibration using a motion tracker

The camera–laser calibration methods involving a checkerboard (calibration plane) [60, 84, 93] have been shown to be reliable and operate on stationary sensors, thus removing any error induced by ego-motion. We therefore obtain our 'ground truth'

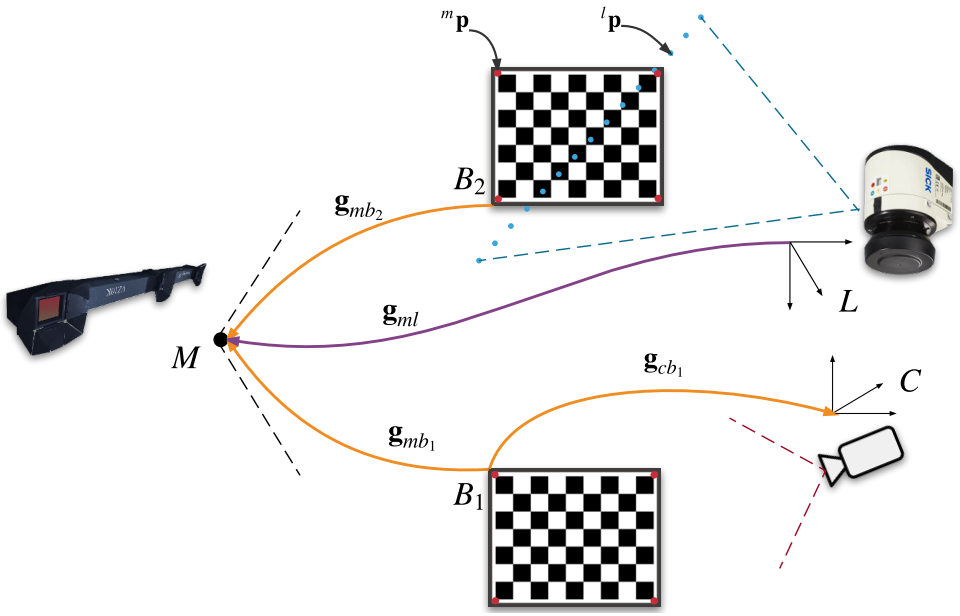


Figure 4.16: Obtaining 'ground truth' calibration using motion tracker. This diagram indicates the notation we use to formulate the calibration solution. The camera and laser do not have overlapping fields of view (indicated by dashed lines), while the motion tracker M overlaps both FoVs.

calibration using an implementation similar to those cited. Calibration is obtained by constraining the laser points found to correspond to the calibration plane, to the plane estimated by the camera. A schematic of the formulation of this calibration problem is shown in Figure 4.1.

One requirement of these methods is that the checkerboard be simultaneously observed by both sensors, allowing for direct data association. We refer the reader to Figures 3.2 and 7.6 of our vehicles which make it clear that this requirement is not met, nor should it be a requirement. To overcome this we introduce a temporary intermediary sensor and reference frame. While an ordinary camera would serve this purpose adequately, we opt for the use of an accurate (sub-0.5mm) motion tracker to estimate the calibration plane. The above problem is subsequently solved independently for laser–tracker and camera–tracker, whilst keeping the tracker stationary.

The goal is to obtain two relative poses $\mathbf{g}_{mc}$ and $\mathbf{g}_{ml}$ that describe camera–tracker and laser–tracker respectively, and infer the the camera–laser pose $\mathbf{g}_{cl} := \ominus\mathbf{g}_{mc} \oplus \mathbf{g}_{ml}$.

Notation is clarified in Figure 4.16 which depicts a typical setup used to solve the camera–laser problem. The fields of view of the camera and laser can be seen to be non-overlapping.

There are two parallel processes required to solve this problem. The tracker is required to have overlapping fields of view with both the laser and camera, and cannot be moved between these processes.

Tracker–Camera: The motion tracker measures the 3D position of active infrared nodes placed around the checkerboard, denoted ${}^{m}\mathbf{p}$ in the figure. We use 12 nodes equally positioned around the edge of the checkerboard, allowing for the estimation of the calibration plane by an overdetermined system. For a single *capture instance*, the positions of the 12 nodes are recorded and a camera snapshot is taken simultaneously. The plane normal with respect to the tracker is estimated as ${}^{m}\boldsymbol{n}$. Using multiple *capture instances*, thus multiple associated normals $\boldsymbol{n}^{i}_{mb_1}$ and $\boldsymbol{n}^{i}_{cb_1}$, the camera–tracker transform can be reliably estimated using [94].

Tracker–Laser: The process for extracting the calibration plane normals with respect to the tracker is the same as in the above step. The set of plane normals can then be used to formulate the coplanar constraint with the laser points registered on the checkerboard [93] with respect to the tracker–laser transform $\mathbf{g}_{ml}$. The final step of obtaining the camera–laser transform is simply $\mathbf{g}_{cl} := \ominus\mathbf{g}_{mc} \oplus \mathbf{g}_{ml}$.

4.8 Concluding remarks

In this chapter we set the context for the remainder of the thesis. We reviewed the literature on 'automatic' calibration and considered that each of the methods reviewed offer varying degrees of autonomy. A data-driven approach is necessary for true autonomy, and fusing of multiple sensor inputs to augment individual processes and to create new ones is beneficial and ubiquitous. The necessity and challenge of supplying accurate inter-sensor calibration thus becomes apparent.

We have also summarised the theoretical and practical elements used to design the camera-laser calibration systems in the following chapters. Of the optimisers, the Nelder-Mead solver is best suited to the information theoretic objective of Section 5.1, while Gauss-Newton is employed to solve the non-linear least-squares problem of Section 5.2.

Algorithm 1 Nelder-Mead algorithm, one iteration

procedure STEP($\mathbf{V}$) ▷ Repeat until convergence
  $\mathbf{V} \leftarrow \mathbf{V}$ sorted ascending w.r.t. $f(\mathbf{v})$
  $\bar{\mathbf{v}} \leftarrow$ mean of $\mathbf{v}_{1\ldots n}$
  $\mathbf{v}_{ref} \leftarrow 2\bar{\mathbf{v}} - \mathbf{v}_{n+1}$
  if $f(\mathbf{v}_{ref}) < f(\mathbf{v}_1)$ **then**
    $\mathbf{v}_{exp} \leftarrow (1+\rho)\bar{\mathbf{v}} - \rho\mathbf{v}_{n+1}$ ▷ $\rho = 2$
    if $f(\mathbf{v}_{exp}) < f(\mathbf{v}_{ref})$ **then**
      $\mathbf{v}_{n+1} \leftarrow \mathbf{v}_{exp}$ ▷ Expansion
    else
      $\mathbf{v}_{n+1} \leftarrow \mathbf{v}_{ref}$ ▷ Reflection
    end if
  else
    if $f(\mathbf{v}_{ref}) < f(\mathbf{v}_n)$ **then**
      $\mathbf{v}_c \leftarrow (1+\psi)\bar{\mathbf{v}} - \psi\mathbf{v}_{n+1}$ ▷ $\psi = 0.5$
      if $f(\mathbf{v}_c) < f(\mathbf{v}_{ref})$ **then**
        $\mathbf{v}_{n+1} \leftarrow \mathbf{v}_c$ ▷ Outside contraction
      else
        $shrink = \text{true}$
      end if
    else
      $\mathbf{v}_{cc} \leftarrow (1-\psi)\bar{\mathbf{v}} + \psi\mathbf{v}_{n+1}$
      if $f(\mathbf{v}_{cc}) < f(\mathbf{v}_{n+1})$ **then**
        $\mathbf{v}_{n+1} \leftarrow \mathbf{v}_{cc}$ ▷ Inside contraction
      else
        $shrink = \text{true}$
      end if
      if $shrink$ **then**
        $\mathbf{v}_j \leftarrow \mathbf{v}_1 + \sigma(\mathbf{v}_j - \mathbf{v}_1)$ for $j = 2 \ldots n+1$ ▷ $\sigma = 0.5$
      end if
    end if
  end if
  $\mathbf{V} \leftarrow (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{n+1})$
  return V
end procedure

Chapter 5

Camera–Laser Calibration

The portable mapping device described in Chapter 2 presents a unique calibration challenge. Particularly with a modular and configurable setup, the process of calibrating needs to be simple and accurate.

We use the term *camera–laser calibration* to describe the general problem of estimating the extrinsic pose parameters that describe a rigidly connected camera and laser scanner. A typical approach is to solve the data association problem by matching correspondences between laser and camera observations.

We therefore study the problem of camera–laser calibration through the lens of end-to-end autonomy. We note efficacy of the automatic methods, but also how reliability can be compromised when dealing with real-world data. This last point is important; a truly automatic system cannot rely on constructed scenes, or environments that have been altered to enhance calibration or feature detection.

We strive for a fully autonomous calibration system that operates using the observations of the robots natural environment, and without the need for manual extraction of co-observed features.

In Section 5.1 we expand on the normalised information distance metric (NID) and apply it to calibration. The formulation of the cost function resembles that used by

[61], which measures data co-dependence using mutual information. We argue for the use of this formulation rather than others presented in the literature. To summarise the advantages of NID; the cost function is robust to changes in appearance and lighting conditions, does not require explicit feature matching, and the normalisation mitigates the effect of the number of visible data points.

Early approaches to calibration match features common to the camera and laser observations explicitly, and in some cases by hand if using real-world observations. In Section 5.2 we propose a method of explicit extraction and matching of point features, yet this is accomplished fully automatically save for the initial calibration estimate. We report the efficacy of the two approaches in the context of mobile ground vehicles. Our tests are conducted on urban roads, pedestrian sidewalks, and in a warehouse, as detailed in Chapter 3.

5.1 Calibration by minimising Normalised Information Distance

This section covers the formulation and optimisation of the objective function we use for calibrating a laser scanner and a camera. We rely on two sets of observations: the grayscale camera image and laser range and reflectance data. The formulation of the objective function uses the information metric *Normalised Information Distance* (NID) to measure and maximise the shared information between the two data streams.

A metric that compares the data directly, such as least squares, assumes that the data occupy a common dynamic range and are proportionally related. We cannot assume this to be the case when use two differing modalities such as laser reflectance and camera luminance, and we demonstrate that this assumption is poor. Image registration using information metrics have shown to be far more robust to noise, occlusion and lighting conditions [20] than the least squares metric. This makes an information metric such as normalised information distance a natural choice for the

calibration problem.

5.1.1 Method

Formulation of the joint histogram

We begin by modelling the observations of the respective sensors as a joint distribution. The model is based on the premise that the sensors at a given point in time observe the same point in $\mathbb{R}^3$, and thus this co-observation contributes to the distribution. The joint distribution of camera and laser intensity data requires the association of a laser point with a pixel location, achieved by projecting the laser point in $\mathbb{R}^3$ to the image space $\mathbb{R}^2$.

Recalling the transform and projection models of Equations (4.3) and (4.4), we express the set of projected points as a function of the pose parameters we seek,

$$\mathbf{x}\left(\mathbf{g}_{CL}\right) = \Pi\left(\mathbf{g}_{CL} \oplus {}^{L}\mathbf{p}\right). \tag{5.1}$$

The timing notation has been dropped for simplicity. A visual representation of the model can be seen in Figure 5.1 where we also define appearance attributes, ${}^{c}\mathcal{I}$ and ${}^{l}\mathcal{I}$ for the camera and laser observations respectively. The latter appearance is simply that observed by the laser scanner for each point, while the former is a function of the position of those points, hence it is dependent on $(u, v) \in \mathbf{x}$. The value of a pixel at position (u, v) is therefore described as ${}^{c}\mathcal{I}(u, v)$.

The underlying data observed by the camera $I(u, v)$, the image, has a derivative of zero ($\frac{\mathrm{d}I}{\mathrm{d}u\mathrm{d}v} = \mathbf{0}$) within a pixel and is otherwise discontinuous at the pixel boundaries. We use bilinear interpolation to estimate the associated luminance at a sub-pixel level, using the operators $\lfloor\cdot\rfloor$ and $\lceil\cdot\rceil$ to denote floor and ceiling functions respectively,

$$
\begin{aligned}
{}^{c}\mathcal{I}(u,v) = \quad & I(u,v)(\lceil u \rceil - u)(\lceil v \rceil - v) \;-\; I(u_{+1},v)(\lfloor u \rfloor - u)(\lceil v \rceil - v) \\
& + I(u_{+1},v_{+1})(\lfloor u \rfloor - u)(\lfloor v \rfloor - v) \;-\; I(u,v_{+1})(\lceil u \rceil - u)(\lfloor v \rfloor - v).
\end{aligned} \tag{5.2}
$$

With the the laser reflectance and associated camera pixel value pair $\{{}^{c}\mathcal{I}(\mathbf{x}), {}^{l}\mathcal{I}\}$, we can generate the joint histogram, which we denote ${}^{cl}\mathbf{P}_{M\times N}$. While algorithmically it is fairly straightforward to generate a joint histogram, we include a formal expression here for completeness. We first denote the operator $[\cdot]^{M} \in \mathbb{Z} > 0$ that indicates the bin to which variable belongs when discretised into M bins. Each bin in the joint histogram will take the value according to

$$
{}^{cl}\mathbf{P}_{ij} = \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{k=1}^{S} \delta(i, [{}^{c}\mathcal{I}_{k}]^{M}).\delta(j, [{}^{l}\mathcal{I}_{k}]^{N}) \tag{5.3}
$$

where we make use of the Kronecker delta function:

$$
\delta(a,b) = \begin{cases} 1 \;:\; a = b \\ \\ 0 \;:\; a \neq b \end{cases}. \tag{5.4}
$$

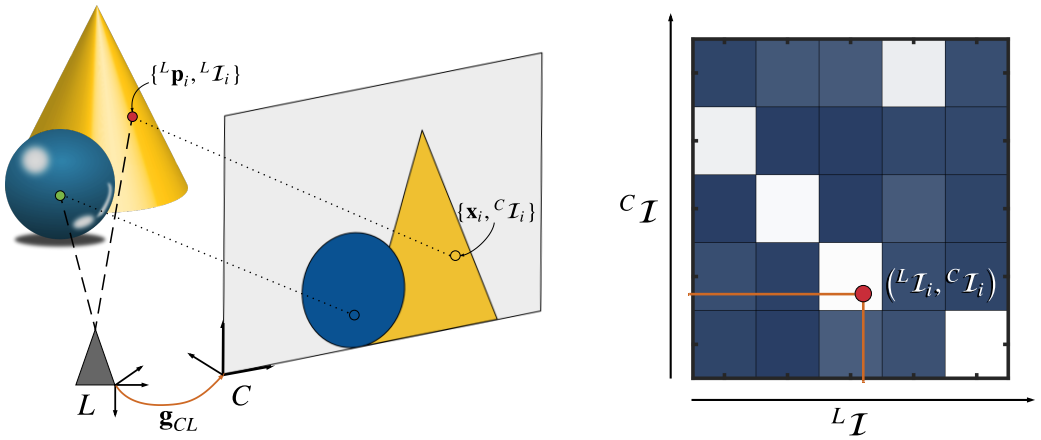
5.1.1.1 Normalised Information Distance for calibration

Recall from Section 4.5 that the NID equation is given as

$$
NID(X,Y) = 1 - \frac{I(X;Y)}{H(X,Y)}.
$$

For a given scene $s \in \mathcal{S}$, and using the joint entropy and mutual information Equations (4.6) and (4.8), the global cost function for calibrating a camera and laser scanner is given as

$$
D^{s}(\mathbf{g}_{cl}) = 1 - \frac{I({}^{c}\mathcal{I}^{s}; {}^{l}\mathcal{I}^{s})}{H({}^{c}\mathcal{I}^{s}, {}^{l}\mathcal{I}^{s})}. \tag{5.5}
$$



a. b.

Figure 5.1: The diagram on the left shows a scene being simultaneously observed by a camera and a laser scanner. The point ${}^L\mathbf{p}_i \in \mathbb{R}^3$ is measured with respect to the laser frame, while $\mathbf{x}_i \in \mathbb{R}^2$ is the projection of that point onto the camera plane. The appearance of the point as viewed by the laser and camera is ${}^L\mathcal{I}$ and ${}^C\mathcal{I}$ respectively. A joint histogram is shown on the right, where the bin falling under the point $({}^L\mathcal{I}, {}^C\mathcal{I})$ is incremented according to Equation (5.3).

We realise that since NID is a metric and $\mathbf{g}_{cl}$ is assumed to be constant within a reasonably small time frame, we can meaningfully sum the cost of multiple scenes. Naturally, we formulate the calibration problem as a minimisation with respect to $\mathbf{g}_{cl}$

$$\hat{\mathbf{g}}_{cl} = \arg\min_{\mathbf{g}_{cl}} \sum_{s \in \mathcal{S}} D^s(\mathbf{g}_{cl}) \tag{5.6}$$

5.1.2 Data pre-processing

Our main focus is the registration of camera pixel luminance to laser reflectance intensity. These attributes are a natural choice because they are both functions of the texture and reflectivity of the surface being measured. However, we have to consider that not only do the sensors report the intensity value in different ranges, the response curves are entirely different. This point is illustrated in Figure 5.2, which shows the joint distribution is poorly constructed as observations are concentrated in only a few bins. The data in this case is used from the scene indicated by the top two

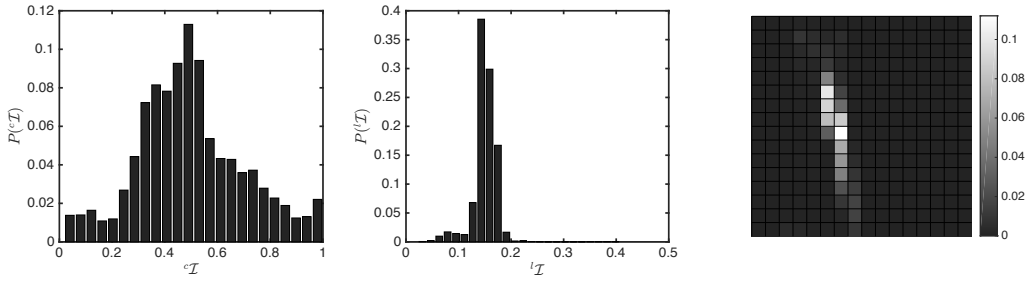


Figure 5.2: Comparison of camera vs. laser raw data distribution. The difference in dynamic range of the two sensors is such that their joint distribution is not well captured by a joint-histogram. From left to right: histogram of camera image, histogram of laser intensity and joint histogram of camera and laser intensity.

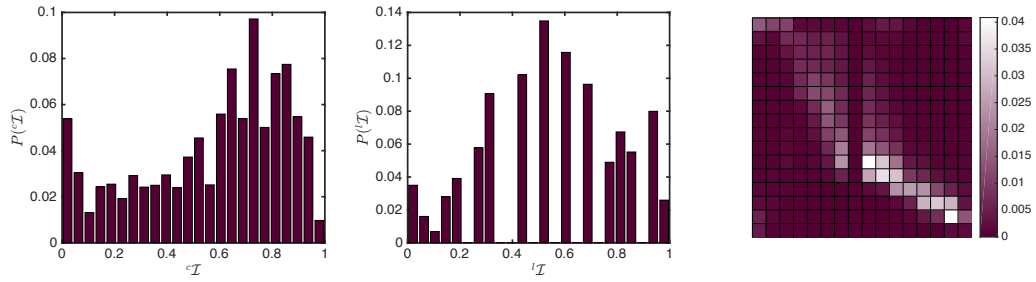


Figure 5.3: Camera and laser data distributions post-equalisation. The distributions don't exhibit the same concentration in certain bins as in Figure 5.2; the joint distribution is spread over the entire histogram and also suggests a value mapping closer to 1:1 (diagonal matrix).

images in Figure 4.2. In this section we investigate how the data can be processed to improve the distribution, and what the overall effect is on the calibration and cost-function convergence characteristics. By 'improving' the distribution, we are effectively increasing the latent entropy of the system. Section 4.5 and particularly Figure 4.7, reminds us why this is the case.

5.1.2.1 Normalisation and Equalisation

Here we attempt to modify the data from each sensor such that they can be reasonably compared. Figure 5.2 shows that the camera image, with correct exposure, produces an image well within the response characteristics of the sensor, save for blowout in the sky. The camera is designed to behave this way. The laser scanner exhibits a response that is concentrated in certain value ranges, leading to the peaky effect we

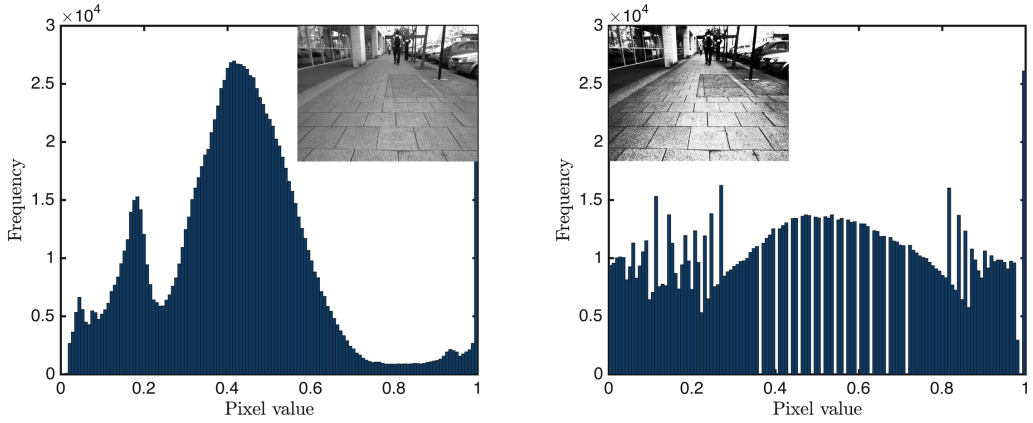


Figure 5.4: Histogram of an image before and after equalisation. For an 8-bit image such as this it is not always possible to redistribute the values to 'fill' all the bins.

see in the distributions.

Normalisation seeks to map the data from both sensors to a common range for a more meaningful comparison. Rather than scale the distribution according to its *min,max* we calculate the standard score for each distribution, thus preventing outliers from dictating the new range. We calculate the new distribution simply using

$$z_i = \frac{x_i - \bar{\mu}}{\sigma} \quad \forall i \ : \ x_i \in X, \tag{5.7}$$

where $\bar{\mu}$ and σ are the mean and standard deviation of X respectively. Importantly, this operation is linear. Histogram equalisation on the other hand modifies the data in a highly non-linear fashion, and thus we proceed with caution. The operation also has the tendency to amplify noise as it naturally increases contrast. Histogram equalisation is used in image processing to increase the contrast of an image by remapping the intensity values such that a nominally monotonic and uniform level curve is obtained. The effect on both the image and the histogram is shown in Figure 5.4, where visually the contrast has been enhanced while the intensity values have been redistributed to create a more uniform histogram.

Histogram equalisation has a significant effect on the entropy of the distributions, and therefore the entropy of the resulting joint distribution. By achieving an even

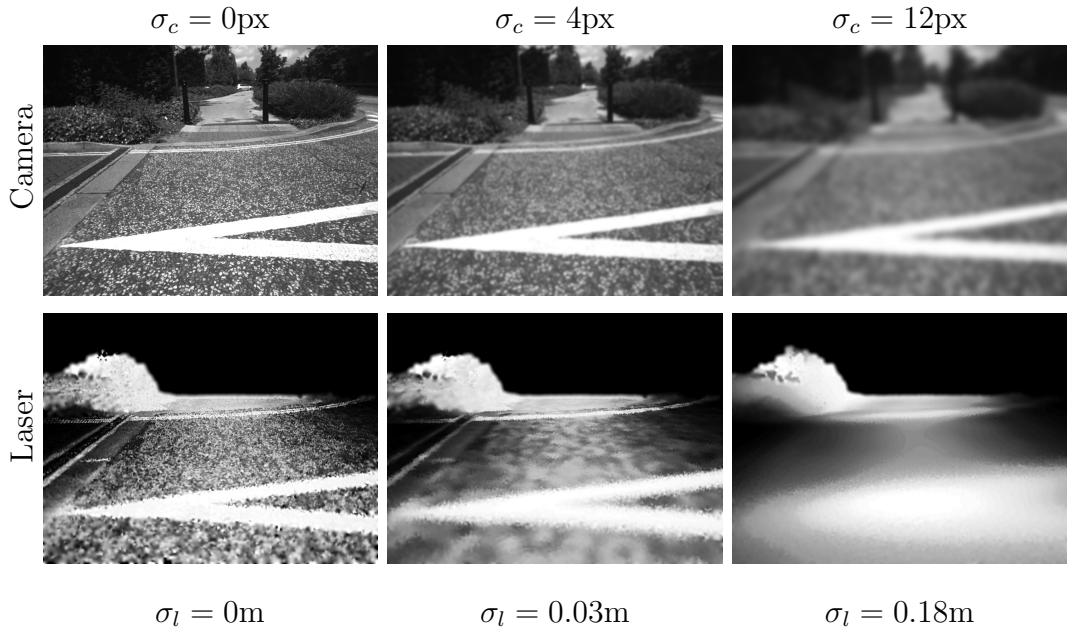


Figure 5.5: Camera and synthetic laser image after Gaussian smoothing. The top row shows the camera image while the bottom row shows the laser projection interpolated to create a dense image. While Gaussian smoothing is effective at removing high-frequency noise, at extremes there can be a severe loss of data.

'spread' of values, the marginal entropies of each modality are effectively matched.

5.1.2.2 Smoothing

Smoothing is a common image enhancement tool [44, 46, 92]. It generally serves two purposes; to reduce high-frequency pixel noise, and to enhance and expand local features such as lines, edges and points. The latter phenomenon is a core component of modern image descriptors that use increasing strengths of Gaussian and Laplacian filters to extract features of different size [9, 46].

Intuitively, edges and regions of high contrast are apt for measuring statistical correlation, yet their locality means that they create discontinuities in the image space. The result is a narrow basin of convergence since features may be initially misaligned as to provide no overlap.

We test the effects of smoothing of both the laser intensity and the camera luminance by inspecting their effects on the basin of convergence. The camera image

is convolved with a Gaussian kernel at varying bandwidths. The laser intensity is smoothed using a spatial Gaussian filter over a neighbourhood of points, weighted according to the distance from the query point. Formally, each laser point α is assigned an appearance value according to

$$
\begin{aligned}
{}^{l}\mathcal{I}_{\alpha} &:= \frac{{}^{l}\mathcal{I}_{\alpha} + \sum_{i=1}^{N} d \cdot {}^{l}\mathcal{I}_{i}}{1 + \sum_{i=1}^{N} d} \\
\text{where} \quad d &= e^{-\frac{(\mathbf{p}\alpha - \mathbf{p}_i)^{\top}(\mathbf{p}\alpha - \mathbf{p}_i)}{2\sigma}}
\end{aligned}
\tag{5.8}
$$

where we choose the N neighbours within a radius $r = 4\sigma$ of $\mathbf{p}_{\alpha}$. The operation is computationally expensive but is run only once. The appearance effects of increasing σ in case of both camera and laser have been prepared in Figure 5.5. The figure also demonstrates the loss of information if too high a smoothing value is chosen.

Of more pertinence are the effects on calibration. On inspection of the cost surface about the point of convergence, we can assess how the blur has affected its convexity and smoothness. The matrix of cost graphs plotted in the directions of X and yaw (Figure 5.6) suggest that a small smoothing of both camera and laser values can widen the basin of convergence — albeit to the detriment of local smoothness about the solution. The third row shows the stark effects of information loss due to over-smoothing.

Remarks on data pre-processing

The varying dynamic ranges of the sensors means that their raw data distributions cannot be captured adequately with a joint histogram. After inspecting the basin of convergence, and comparing the effects of mean-normalisation and histogram equalisation, we recommend histogram equalisation of both the camera and laser intensity values. Using the standard score however also smooths discontinuities in the cost surface, but combining it with the histogram equalisation seemed detrimental.

Adding Gaussian smoothing to the camera and laser intensity data had marginal

effects. Choosing too high a smoothing parameter resulted in loss of information, thus we recommend choosing a small smoothing factor, to expand local features such as edges. Mutual information is shown to be very robust to sensor noise [20], and as such, smoothing should have little effect in this regard.

For the experiments in the following section, an image smoothing width of $\sigma_{camera} = 4$ was used[1]. Both camera and laser intensity data are histogram equalised on a per-image and per-swathe basis respectively.

5.1.3 Results: NID

This section reports the results of camera–laser calibration using NID running on ORI datasets, notably those of the GEM, LEAF and *NABU*. As other authors have noted, the appearance of the scene can have a drastic effect on the performance and reliability of the registration process [52, 61, 88, 95]. This problem is overcome using our scene selection method described in Chapter 6. The results we report here are predicated on having selected scenes that reliably allow for registration according to this method.

Calibration of a camera–laser pair

We evaluate the calibration algorithm on the four datasets, each representing different environments. The results for calibration presented here are of push-broom lasers and the front facing stereo-camera on all vehicles. Calibration was done between the laser and left and right cameras of the stereo unit respectively, assuming true and known inter-camera calibration, yet calibration estimates are reported with respect to the left camera frame only. The calibration values pertaining to $\hat{\mathbf{g}}_{c_2l_1}$ are reported as $\mathbf{g}_{c_1c_2} \oplus \hat{\mathbf{g}}_{c_2l_1}$ for ease of comparison, assuming the camera–camera calibration $\mathbf{g}_{c_1c_2}$ to be known and accurate.

In each case, 20 scenes were sampled at random from the 200 top-scoring scenes

[1]This value is applicable to camera sensors of between one and two megapixels

Table 5.1: Calibration Results: GEM Bumblebee–Laser1 using 20 scenes

	X	Y	Z	ϕ'_x	ϕ'_y	ϕ'_z	$\|\boldsymbol{\phi}\|$
$\mathbf{g}_{truth}$	-2.580	0.113	0.336	0.766	0.013	0.643	3.134
$\hat{\mathbf{g}}_{c_1 l_1}$	$-2.532^{\pm 0.10}$	$0.117^{\pm 0.01}$	$0.374^{\pm 0.01}$	0.749	0.002	0.663	$3.125^{\pm 0.02}$
				$\hat{\boldsymbol{\kappa}} =$	921		
$\hat{\mathbf{g}}_{c_2 l_1}$	$-2.591^{\pm 0.08}$	$0.127^{\pm 0.01}$	$0.381^{\pm 0.01}$	0.765	0.006	0.644	$3.122^{\pm 0.04}$
				$\hat{\boldsymbol{\kappa}} =$	1468		

Table 5.2: Calibration Results: LEAF Bumblebee–Laser1 using 20 scenes

	X	Y	Z	ϕ'_x	ϕ'_y	ϕ'_z	$\|\boldsymbol{\phi}\|$
$\mathbf{g}_{truth}$	2.011	0.212	1.118	0.001	-1.000	0.002	1.517
$\hat{\mathbf{g}}_{c_1 l_1}$	$1.949^{\pm 0.09}$	$0.218^{\pm 0.01}$	$1.035^{\pm 0.01}$	-0.013	-1.000	0.001	$1.509^{\pm 0.09}$
				$\hat{\boldsymbol{\kappa}} =$	985		
$\hat{\mathbf{g}}_{c_2 l_1}$	$1.975^{\pm 0.08}$	$0.229^{\pm 0.01}$	$1.039^{\pm 0.01}$	-0.014	-1.000	-0.001	$1.509^{\pm 0.09}$
				$\hat{\boldsymbol{\kappa}} =$	1648		

(see Chapter 6) and the calibration evaluated starting from 40 different random seeds, $\mathbf{g}_0 = \mathcal{N}(\mathbf{g}_{truth}, \mathbf{I}\sigma)$ to obtain a variance using a Monte-Carlo approach. The initial seed variance was chosen as $\sigma^2 = 0.1$m. The translation components are reported in meters while the angles are reported as a unit vector $\boldsymbol{\phi}' = (\phi'_x, \phi'_y, \phi'_z)$ representing the axis of rotation, and magnitude $\|\boldsymbol{\phi}\|$ in radians. The standard deviation is indicated using a superscript $\pm$.

We model the angles using a von Mises-Fisher distribution, described in Section 4.7. The deviations in magnitude of the rotations are assumed to be sufficiently small that they can be mapped to a linear space, thus mean and variance of the angle of rotation use the standard formulae. The results of testing on the GEM, RobotCar and NABU 3 are presented in Tables 5.1–5.4.

It is worth bearing in mind that the variance parameter used to generate the initial seeds was set to $\sigma^2 = 0.1$ thus any variance in the parameters below this has indicated consistent convergence. Conversely, values greater than $\sigma^2 = 0.1$ will indicate that the problem is poorly constrained in that parameter. For example, the standard

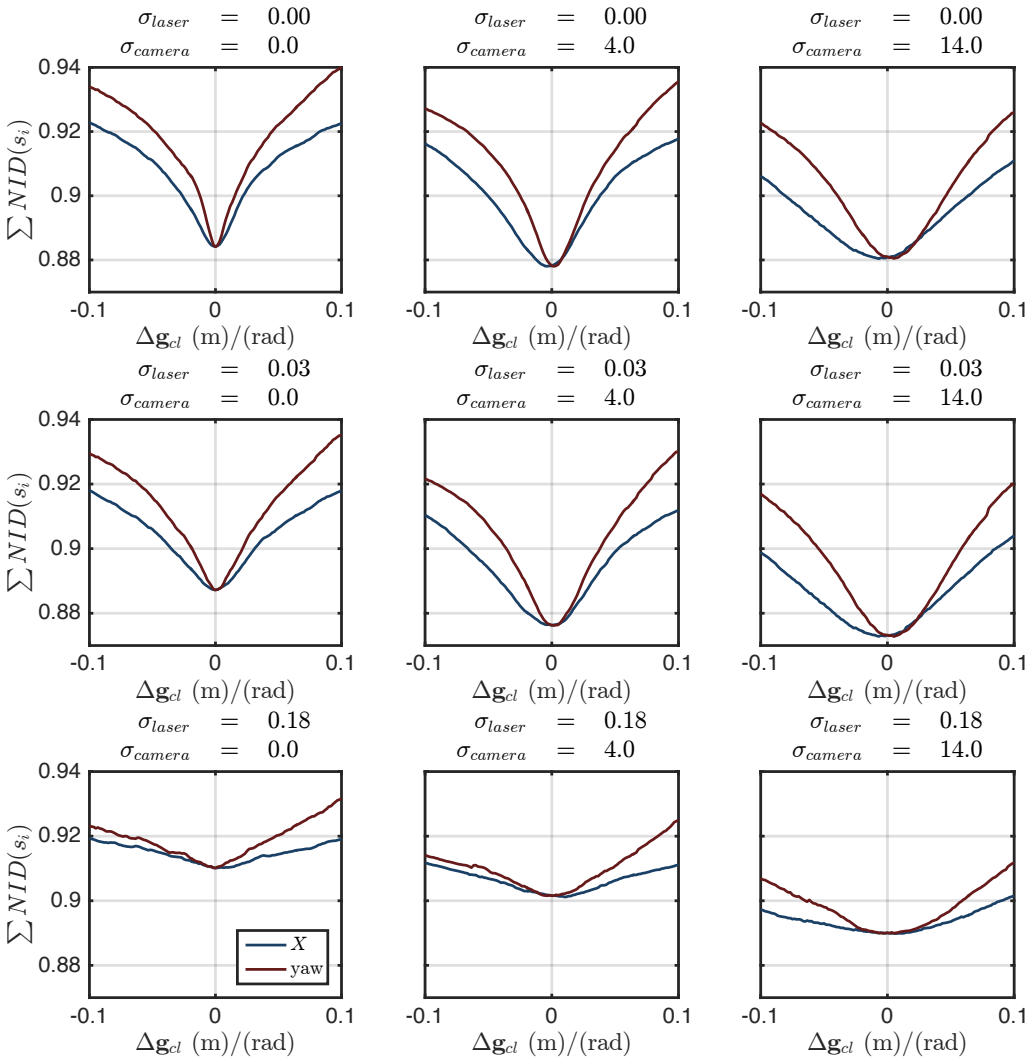


Figure 5.6: Effects of Gaussian smoothing of camera and laser data on the NID. From left to right, the camera image is blurred with increasing sigma. From top to bottom, the laser intensity is smoothed with increasing sigma. Small amounts of smoothing in both laser and camera intensity widen the basin of convergence, *row 2, column 3*, while over-smoothing reduces information (*row 3*) and will adversely affect convergence.

Table 5.3: Calibration Results: LEAF Bumblebee–Laser2 using 20 scenes

	X	Y	Z	ϕ'_x	ϕ'_y	ϕ'_z	$\|\boldsymbol{\phi}\|$
$\mathbf{g}_{truth}$	-2.504	0.150	1.048	-0.719	-0.001	-0.695	3.141
$\hat{\mathbf{g}}_{c_1 l_2}$	$-2.607^{\pm 0.09}$	$0.101^{\pm 0.05}$	$1.160^{\pm 0.12}$	-0.722	0.018	-0.692	$3.147^{\pm 0.05}$
				$\hat{\boldsymbol{\kappa}} =$	1116		
$\hat{\mathbf{g}}_{c_2 l_2}$	$-2.549^{\pm 0.07}$	$0.117^{\pm 0.02}$	$1.154^{\pm 0.05}$	-0.700	0.026	-0.714	$3.126^{\pm 0.03}$
				$\hat{\boldsymbol{\kappa}} =$	826		

Table 5.4: Calibration Results: *NABU 3*–Laser1 using 20 scenes

	X	Y	Z	ϕ'_x	ϕ'_y	ϕ'_z	$\|\boldsymbol{\phi}\|$
$\mathbf{g}_{truth}$	-0.009	-0.180	-0.001	0.140	0.988	0.061	1.264
$\hat{\mathbf{g}}_{c_1 l_1}$	$-0.073^{\pm 0.08}$	$-0.136^{\pm 0.1}$	$0.002^{\pm 0.07}$	0.144	0.988	0.059	$1.267^{\pm 0.12}$
				$\hat{\boldsymbol{\kappa}} =$	585		
$\hat{\mathbf{g}}_{c_2 l_1}$	$-0.122^{\pm 0.13}$	$-0.176^{\pm 0.06}$	$0.107^{\pm 0.14}$	0.162	0.985	0.055	$1.247^{\pm 0.11}$
				$\hat{\boldsymbol{\kappa}} =$	209		

deviation of the X parameter indicates high uncertainty in all testing cases. The initial axis concentration had an initial starting value of $\boldsymbol{\kappa} \approx 450$, thus concentration has also improved after convergence.

Fusing multiple scenes

The formulation of the cost function naturally lends itself to combining the cost over multiple scenes. We obtain a Monte-Carlo standard deviation for each scene subset. To achieve sufficient data sampling and randomness, we follow Algorithm 2. To summarise, for each size N_i^s, we choose N_i^s scenes at random from $\mathcal{S}$ and run the optimiser starting from 8 different random seeds sampled from a normal distribution near the solution. This process is then repeated with a new random subset 5 times, limited by computational time.

Algorithm 2 Monte-Carlo solutions of $\hat{\mathbf{g}}$

procedure SolveOverAndOver
  for $N_i^s \leftarrow 1, 40$ **do**
    for $j \leftarrow 1, 5$ **do**
      $s \leftarrow \binom{\mathcal{S}}{N_i^s}$ ▷ thus $s \subset \mathcal{S}$
      for $k \leftarrow 1, 8$ **do**
        $\mathbf{g}_0 \leftarrow \mathcal{N}(\mathbf{g}_{truth}, \mathbf{I}\sigma)$
        $\mathbf{g}_{jk}^i \leftarrow$ solution to Equation (5.6) over s, starting at $\mathbf{g}_0$
      end for
    end for
  end for
end procedure

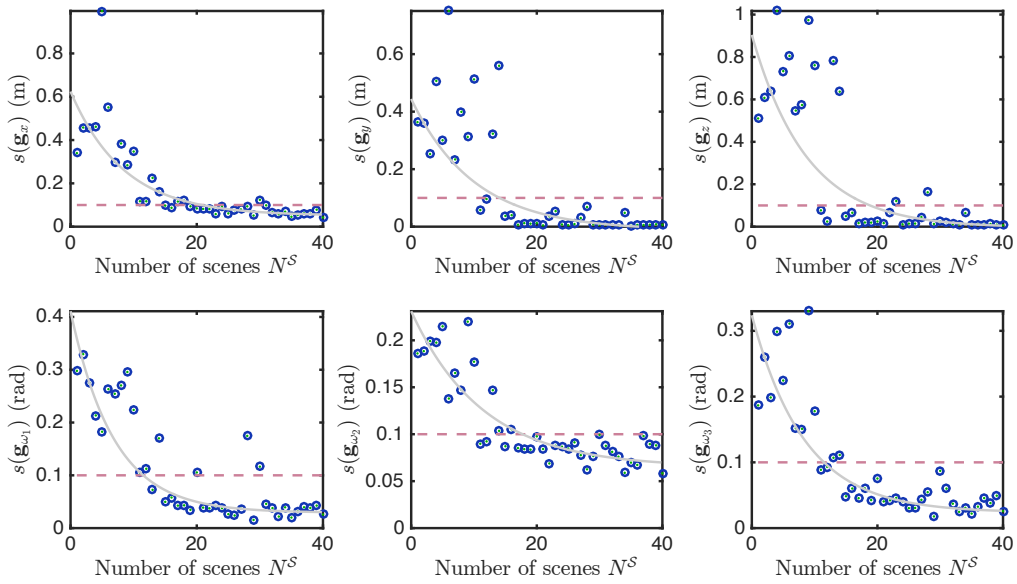


Figure 5.7: Plots showing the effect of adding scenes to the optimisation. In line with other work, adding more scenes lowers the standard deviation of the solution, making us more certain. The grey line is a decaying trend function fitted to the data. The standard deviation was calculated using the Monte-Carlo method, whose initial seed standard deviation is indicated by the horizontal red line, $\sigma^2 = 0.1$

The results for calibration on an Oxford 10km LEAF dataset are plotted in Figure 5.7 and, consistent with other research, show that increasing the number of scenes used significantly and quickly reduces the variance of each parameter. We note that calibration tends to diverge with insufficient data. We remind the reader at this point that the dataset from which we have drawn a sample consists of the top 10% of scenes (approximately 200) ranked using our method in Chapter 6. This is to ensure convergence as the vast majority of the scenes we tested cause divergence.

5.1.4 Discussion

In this section we discuss the results presented above, and note implementation details particular to our NID calibration.

All but the *NABU 3* show evidence for successful calibration. The parameters presented in Table 5.4 have a high variance and low $\hat{\boldsymbol{\kappa}}$ falling outside of the variance used to generate the initial seed. Since the laser–camera configuration of the *NABU 3*

is very similar to the other platforms, we can deduce that the nature of the observed warehouse is such that it is not conducive to calibration using the NID method. The automatic calibration method we propose in Section 5.2 shows successful calibration of the *NABU*.

Calibration performance

Camera–laser calibration using NID has shown to be effective only under certain conditions. Authors investigating calibration using natural scenes assert that summing the cost over multiple scenes sufficiently improves calibration accuracy and variance. We argue that this is true only in part, and the inclusion of offending scenes can cause calibration to fail. While the results presented in Figure 5.7 do support those assertions, we are obliged to disclaim that the scenes we used for experimentation are not arbitrary; they have been diligently selected using the method we propose in Chapter 6.

We can speculate that this difference in outcomes is due to different hardware and configurations. The push-broom laser requires accurate ego-motion estimates to generate a metrically correct point cloud swathe. A 3D snapshot taken by a 3D laser scanner such as a Velodyne with the vehicle at rest would not suffer from the same uncertainties.

The most likely factor contributing to the poor results of the *NABU 3* is that the warehouse environment differs considerably from the urban outdoors. The photograph in Figure 3.7 is representative of this particular dataset; featureless floors and irregular and non-planar vertical structures. The warehouse dataset is 690m, compared to the 4km and 10km of the GEM and LEAF datasets respectively. It is possible that given the nature of the environment, there is insufficient data to procure good scenes for calibration; low variability in appearance reduces the chances of acquiring feature-rich data.

Point cloud swathe creation

NID calibration depends on a swathe of 3D points that cover the world as seen by the camera. A swathe is a 'short map', or local representation of the robots environment.

We select a map length of 12m (typically the distance traversed by the vehicle) as we found this to provide sufficient overlap of laser points with the camera's field of view whilst not suffering from high point concentrations near the horizon; a phenomenon highlighted in [76]. Laser points nearer the vehicle are also less prone to errors propagated through the ego-motion, and fall on surfaces which are captured in higher detail due to their being nearer the sensor.

Initial calibration estimate

The experiment measuring the variance of the calibration estimates shows that convergence happens reliably when the initial error is within $\pm$10cm and $\pm 6^{\circ}$ at one standard deviation. We were able to hand-measure the system to well within this range.

The RobotCar and the GEM have both been retrofitted with sensors so their metric configuration is initially unknown. The sensors on the *NABU 3* have been positioned according to CAD drawings and precise manufacturing methods, thus providing near-perfect calibration estimates. In reality, even small biases in sensor centre can cause significant and visible miscalibration thus still requiring further optimisation.

Optimisation using Nelder-Mead method

We opted for a non-gradient based optimisation method due to concerns of the cost function being non-smooth. The trade-off between speed and robustness is one we can afford to make. The cost surfaces presented in Figure 5.6 appear to be monotonic, yet inspection of the derivative illustrates that this is not the case. Evaluating the

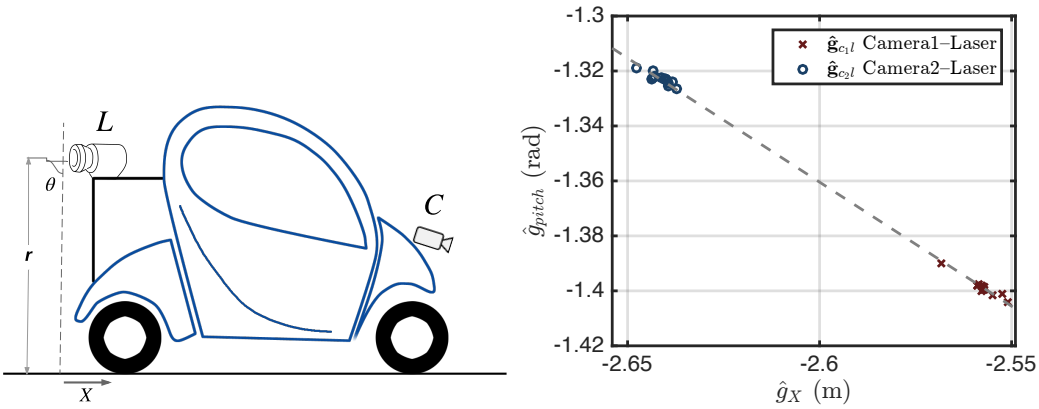


Figure 5.8: Possible source of uncertainty due to laser scanning plane being positioned at approximately 90° to the X direction, coinciding with point observations lying predominantly on the ground plane. Laser points relative to the vehicle frame would exhibit proportional shift for small changes in X or in *pitch*, as $\Delta X \approx \Delta\theta \cdot r$. The right-hand plot presents 15 solutions of X and *pitch* of the results plotted in Figure 5.7.

numeric derivative of the cost function computed with small $\delta\mathbf{g}$, as plotted in Figure 4.11, it is evident that the cost function is in fact highly non-smooth. For this reason we continue to recommend a non-gradient based optimisation such as the Nelder-Mead method, or other methods that make use of a simplex gradient [37]. Our experimentation has found the use of Nelder-Mead method to consistently locate potentially global minima.

Sources of uncertainty in X and *pitch*

Inspecting the calibration error results of Figure 5.7, we notice unusually large uncertainty in both the X and *pitch* parameter. On further investigation, we discover that there is some 'slack' at the intersection of the X and *pitch* parameters. The plot in Figure 5.8 shows the plotted solution of these parameters with respect to multiple calibration estimates of both cameras in the stereo unit of the GEM.

A plausible hypothesis is that the derivatives of the cost function with respect to those parameters are near-proportional — if perhaps opposite in sign. The schematic in Figure 5.8 demonstrates how this could manifest: a small shift in the laser angle θ

would induce an apparent and proportional shift in X of points located on the ground plane. The lower the height, r, the greater the potential for there to be variability in *pitch*. The plot in the figure shows the distribution of calibration estimates of X and *pitch* for each camera respectively, which gives evidence of a proportionality between the parameters.

This same effect is observed in the point feature-based calibration method covered in the following section.

5.2 Calibration by automatic 2D point feature matching

We can broadly categorise calibration methods covered in Section 4.1 as (1) automatic feature extraction with known scene geometry, (2) manual data association on real-world scenes, or (3) solving a global metric with real-world scenes.

This section proposes a further method: automatic feature extraction and matching in real-world scenes. We describe how we use a common point feature detector to automatically perform the association between a point in $\mathbb{R}^2$ image space and a 3D laser point.

This method relies on a *dense* synthetic laser image, such as that generated using the methods in Sections 4.3 and 4.4, with the additional step of using interpolation to fill in missing values. Therefore, the algorithm is conditioned on the availability of a densely constructed point cloud of the scene, and reflectance values associated with each laser point. A perhaps stronger requirement is that the intensity map between camera pixel luminance and laser reflectance is monotonic. That is to say that laser intensity is at least proportional to pixel luminance. As with most calibration methods, a reasonable initial seed must be supplied. The schematic in Figure 5.1 offers a visual representation of the generation of the sparse synthetic image ${}^{L}\mathcal{I}$.

The application of photogrammetric features to synthesised laser images has been

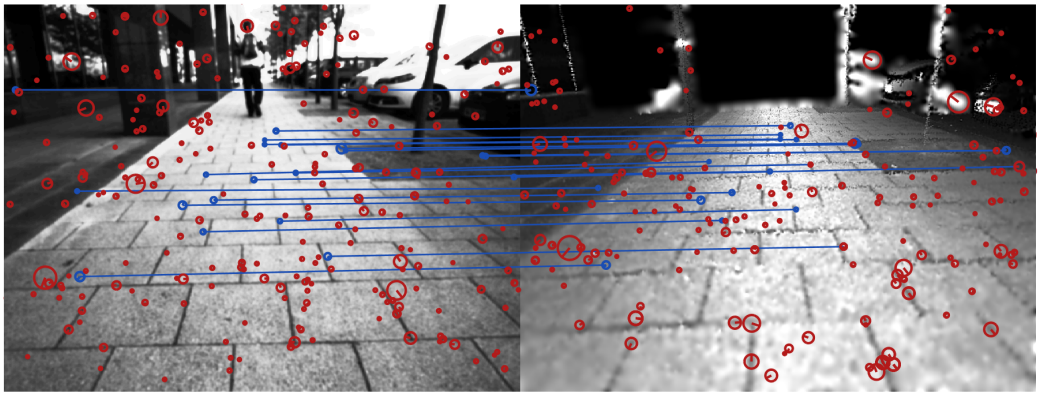


Figure 5.9: SIFT features matched between camera image (left) and laser image. Inliers are considered to be features matched within a 40 pixel radius. Outliers indicated in red.

applied to registering photographs with terrestrial laser scans [14, 50], or to register multiple laser scans [91]. The terrestrial laser scanners are high-fidelity devices, similar to the Riegl in Figure 2.2, capable of dense and accurate scene reconstructions. We apply the same principles to the feature extraction and registration using a low-fidelity Hokuyo UTM-30LX, a planar laser scanner.

5.2.1 Method

Visual odometry or camera–camera calibration attempts to solve the rigid transform between two image instances. A common way to achieve this is to identify features that exist in both images and find the camera–camera transform that best satisfies the constraints imposed by the epipolar geometry of a pin-hole camera model [31].

The method we propose here is a variation of the above. The two images in question are the camera image, and the synthetic laser image created at the same time instant. One could imagine Figure 5.9 being taken from a stereo visual-odometry paper, whereby extracted point features are successfully matched across images. Note that these two images occupy the same frame of reference; the transform between the two image centres is the identity. Instead we solve for the camera–laser transform on which the synthetic laser image depends.

The calibration algorithm presented here uses VLFeat's [85] implementation of SIFT (Scale Invariant Feature Transform) [46], and although not tested, would likely work with others such as BRIEF [16], SURF [8] or ORB [69].

The purpose of matching point features is to create a hard association between a point in the space of the camera image, and the corresponding laser point. Both represent the same 3D point in the world. The process of this is as follows.

We are presented with the two images; ${}^{L}\mathcal{I}(\mathbf{g}_0), {}^{C}\mathcal{I} \subset \mathbb{R}^2$. The former is the dense synthetically generated laser image using the initial calibration estimate $\mathbf{g}_0$. Both have the same dimensionality and the camera image uses a single channel consisting of a grayscale value. Both camera and laser data streams undergo histogram equalisation prior to SIFT feature extraction.

The sets of SIFT descriptors, $\{{}^{L}\boldsymbol{x}, {}^{L}\boldsymbol{F}\}$ and $\{{}^{C}\boldsymbol{x}, {}^{C}\boldsymbol{F}\}$, for the laser and camera image respectively are extracted using [85]. The descriptors ${}^{L}\boldsymbol{F}$ are located at ${}^{L}\boldsymbol{x} \subset \mathbb{R}^2$. An association is made between two neighbouring descriptors $\{{}^{L}\boldsymbol{f}_n \in {}^{L}\boldsymbol{F}, {}^{C}\boldsymbol{f}_n \in {}^{C}\boldsymbol{F}\}$ if the distance between them is less than the distance between both descriptors and all other descriptors multiplied by a threshold ratio [46], chosen as $K = 0.7$. Formally, a match is determined if:

$$\|{}^{L}\boldsymbol{f}_n - {}^{C}\boldsymbol{f}_n\| < \min(\|{}^{L}\boldsymbol{f}_n - \boldsymbol{f}\|, \|{}^{C}\boldsymbol{f}_n - \boldsymbol{f}\|) \cdot K \;:\; \forall \boldsymbol{f} \in \{{}^{L}\boldsymbol{F} \cap {}^{C}\boldsymbol{F}\} - \{{}^{L}\boldsymbol{f}_n\} - \{{}^{C}\boldsymbol{f}_n\}. \tag{5.9}$$

Furthermore, we reject a match if the pixel locations differ significantly; a match must satisfy

$$\|{}^{L}\boldsymbol{x}_n - {}^{C}\boldsymbol{x}_n\| < T, \tag{5.10}$$

for which we choose $T = 40$ pixels to allow for initial calibration error.

Referring to Figure 5.1, recall that the location of a projected laser point is denoted $\mathbf{x}_i$. We associate a matched point descriptor ${}^{L}\boldsymbol{f}_n$ with a laser point by finding its nearest neighbour in image space, thus satisfying $\min_i \|{}^{L}\boldsymbol{x}_n - \mathbf{x}_i\|$. A nearest-neighbour

search can be accomplished efficiently using a k-dimensional tree.

The deduced association of image feature location to laser point $\{{}^{C}\boldsymbol{x}_n, \mathbf{x}_i\} \subset \mathbb{R}^2$, determined only once using $\mathbf{g}_0$ as an initial guess, allows us to formulate a constraint which aims to minimise the reprojection error. Notice that the projection of the laser point is a function of the calibration parameters, while the location of the camera image feature ${}^{C}\boldsymbol{x}_n$ is treated as constant. Over all positive associations of i and n, and using Equations (4.2 – 4.3), the objective function is formulated as

$$\hat{\mathbf{g}}_{cl} = \arg\min_{\mathbf{g}_{cl}} \sum_{n,i} L_\alpha\big({}^{C}\boldsymbol{x}_n, \mathbf{x}_i(\mathbf{g}_{cl})\big). \tag{5.11}$$

Here we use the Huber loss kernel L_α to mitigate the effect of false matches [34], and is defined as

$$L_\alpha(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2 & |y - f(x)| < \alpha \\ \alpha|y - f(x)| - \frac{1}{2}\alpha^2 & \text{otherwise.} \end{cases} \tag{5.12}$$

The solution is found using the Levenberg-Marquardt minimiser described in Section 4.6, using a value of $\alpha = 12$ pixels. The results follow in the next section.

5.2.2 Results: SIFT point features

We test the calibration on the Milton Keynes dataset recorded using the GEM, and a warehouse dataset collected using the *NABU 3*. This section presents the results of calibrating the push-broom laser scanner to each camera of the stereo-camera unit. This laser–camera configuration on the GEM and the *NABU 3* are principally the same; a 2D laser scanner is used to generate point cloud swathes using motion estimates from the stereo camera's visual odometry.

We also present results pertaining to the extraction and matching of point features, which is greatly dependent on the quality of the synthetic laser image

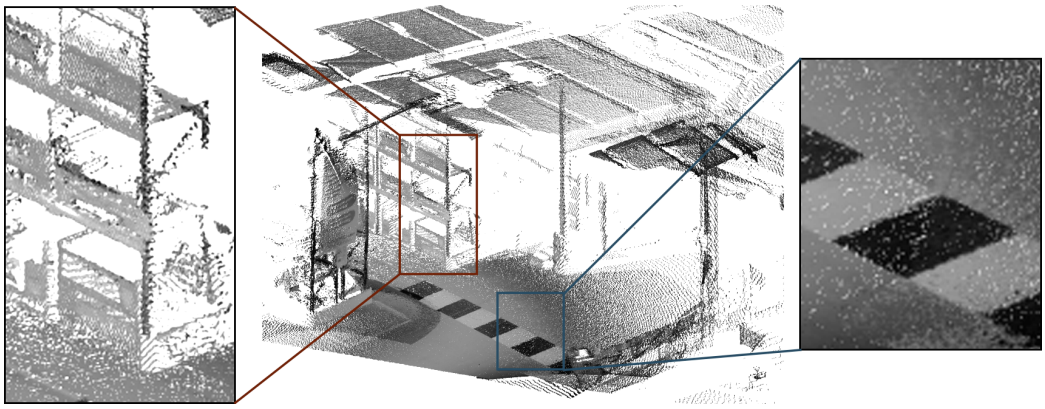


Figure 5.10: Point cloud segment of a warehouse generated by *NABU 3*. The segment merges point clouds swathes generated by two push-broom lidars, both calibrated using automatic point feature matching. The enlarged detail shows the point clouds are strongly aligned.

5.2.2.1 Calibration performance

We present the calibration performance according to three measures as discussed in Section 4.7.1; accuracy and precision with respect to 'ground truth' (1) and camera–camera calibration (2), and confirmation by visual inspection (3).

Calibration parameters of the camera–laser of the GEM and *NABU 3* are presented in Tables 5.5 and 5.6 in the same manner as the tables in Section 5.1.3, with ground truth available on the GEM. In each case, 155 successful SIFT feature matches were used to constrain the problem. The mean solutions and standard deviations are presented in the tables, obtained by choosing 15 sets of 155 points and solving in each case, equivalent to reading from the latter end of the x-axis in Figures 5.11 and 5.12. The result of $\hat{\mathbf{g}}_{c_2l_1}$ is presented with respect to Camera 1, presuming that the Camera1–Camera2 calibration is perfect and thus constant. The bottom row therefore shows $\mathbf{g}_{c_1c_2} \oplus \hat{\mathbf{g}}_{c_2l_1}$, which provides an easier comparison with $\hat{\mathbf{g}}_{c_1l_1}$. The translation components are presented in meters, while the angles are reported using the axis-angle representation covered in Section 4.7.1.

The second mode of evaluation uses the two solutions, $\hat{\mathbf{g}}_{c_1l_1}$ and $\hat{\mathbf{g}}_{c_2l_1}$, to infer the camera–camera transform of the stereo-camera unit. We presume that the calibration

Table 5.5: Calibration results (point features): *NABU 3* Bumblebee–Laser1 using 155 points

	X	Y	Z	ϕ'_x	ϕ'_y	ϕ'_z	$\|\boldsymbol{\phi}\|$
$\mathbf{g}_{truth}$	-0.009	-0.180	-0.001	0.140	0.988	0.061	1.264
$\hat{\mathbf{g}}_{c_1 l_1}$	$-0.014^{\pm 0.01}$	$-0.147^{\pm 0.02}$	$-0.014^{\pm 0.01}$	0.134	0.989	0.066	$1.270^{\pm 0.01}$
				$\hat{\boldsymbol{\kappa}} =$	19179		
$\hat{\mathbf{g}}_{c_2 l_1}$	$-0.014^{\pm 0.02}$	$-0.167^{\pm 0.02}$	$0.001^{\pm 0.01}$	0.132	0.990	0.049	$1.256^{\pm 0.01}$
				$\hat{\boldsymbol{\kappa}} =$	15381		

Table 5.6: Calibration results (point features): GEM Bumblebee–Laser1 using 155 points

	X	Y	Z	ϕ'_x	ϕ'_y	ϕ'_z	$\|\boldsymbol{\phi}\|$
$\mathbf{g}_{truth}$	-2.580	0.113	0.336	0.766	0.013	0.643	3.134
$\hat{\mathbf{g}}_{c_1 l_1}$	$-2.610^{\pm 0.13}$	$0.125^{\pm <0.01}$	$0.392^{\pm 0.04}$	0.781	0.012	0.624	$3.130^{\pm <0.01}$
				$\hat{\boldsymbol{\kappa}} =$	728		
$\hat{\mathbf{g}}_{c_2 l_1}$	$-2.642^{\pm 0.04}$	$0.122^{\pm <0.01}$	$0.385^{\pm 0.01}$	0.786	0.017	0.619	$3.132^{\pm <0.01}$
				$\hat{\boldsymbol{\kappa}} =$	6750		

supplied by the manufacturer has a significantly lower error than the calibration we are inferring. The error is calculated as follows

$$\mathbf{g}_{err} = \hat{\mathbf{g}}_{c_1 l_1} \ominus \hat{\mathbf{g}}_{c_2 l_1} \ominus \mathbf{g}_{c_1 c_2}. \tag{5.13}$$

Figures 5.11 and 5.12 show the distributions of calibration error for the *NABU 3* and GEM respectively. The mean error and standard deviation for each parameter is reported for differing number of point feature matches. We report the error using Euler angles due to the angles being small.

The calibration results of the *NABU 3* using warehouse data are precise to between 1 and 2 centimetres, with very high consistency in angles. The *NABU 3* equipped with two laser scanners poses a particularly challenging calibration problem, which would expose clear visual anomalies if the two lasers are not mutually well-calibrated with respect to the centre of motion.

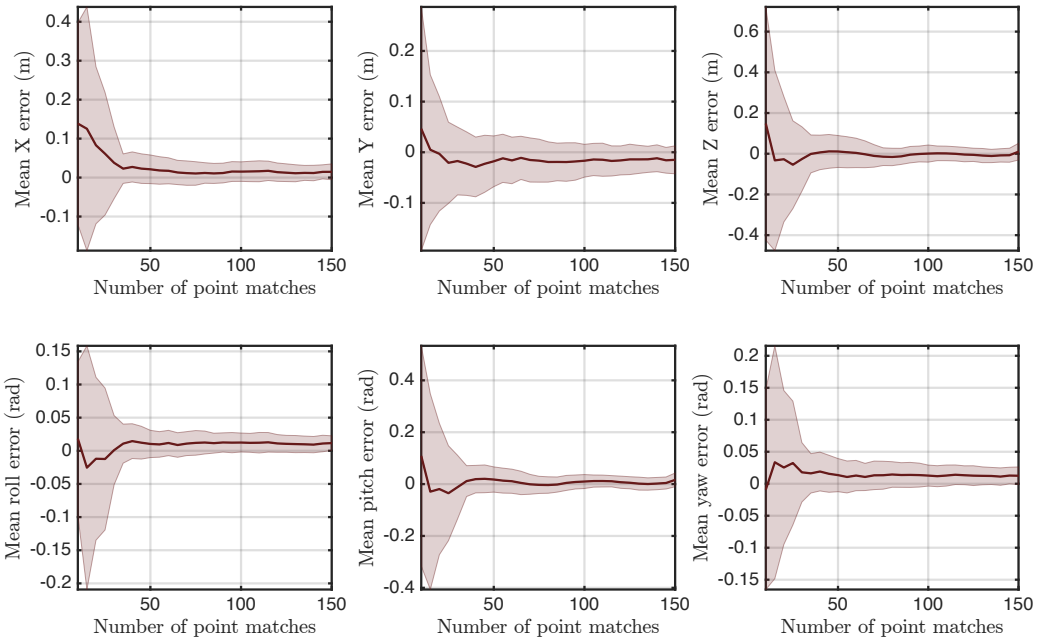


Figure 5.11: Graphs of camera–laser calibration error, *NABU 3*. The mean error for each parameter is indicated by the solid line. The shaded regions indicate $\pm 1\sigma$. Increasing the number of matched points used in the optimisation is shown to decrease error and variance of the parameters. Error for using fewer than 20 points is not shown due to frequent non-convergence.

The point cloud of the warehouse in Figure 5.10 is a combination of two swathes generated using two laser scanners in push-broom configuration. The appearance of the points in the figure is determined by the reflectance values reported by the laser scanners. Each of the scanners is calibrated independently with respect to the stereo-camera using the point-feature method. The insets show that there is no visible incongruity where the two swathes overlap; the black-and-white path is not disjointed, and the fine detail of the shelf uprights is consistent.

5.2.2.2 Matching performance and reliability

As a preprocessing step, we applied a histogram equalisation to both the camera and laser data. While SIFT features offer a robustness to appearance change, we found that more feature matches were discovered when the dynamic ranges of both sensors were normalised.

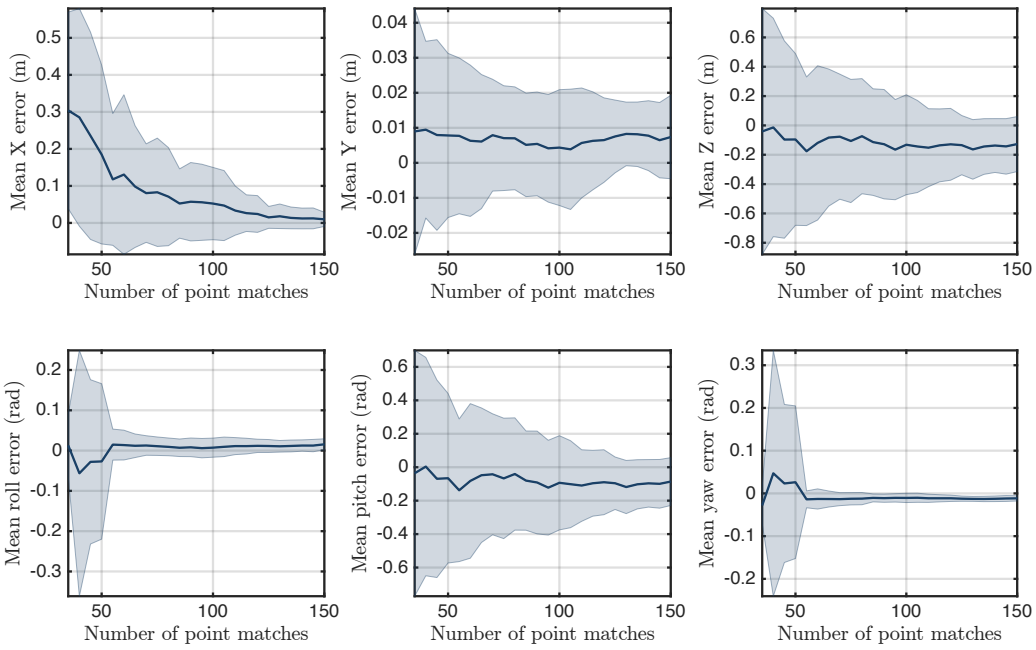
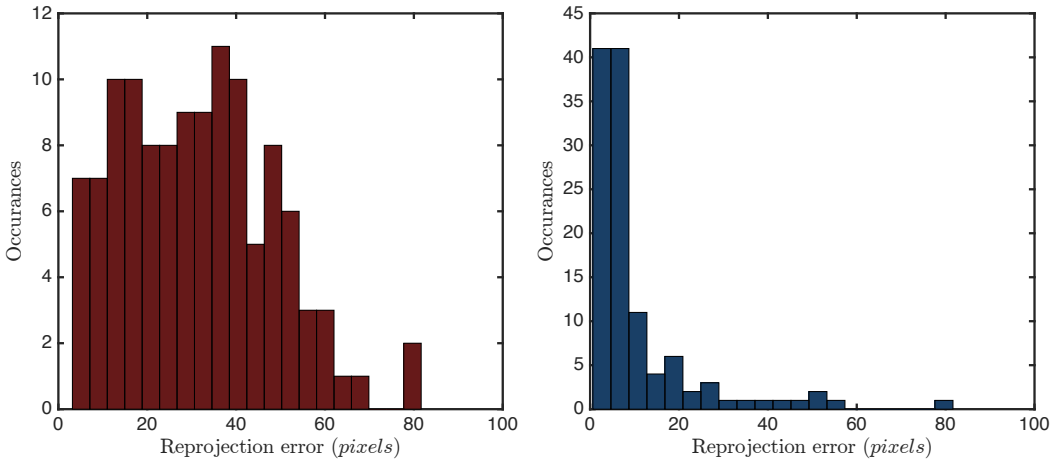


Figure 5.12: Graphs of camera–laser calibration error, GEM. The mean error for each parameter is indicated by the solid line. The shaded regions indicate $\pm 1\sigma$. Increasing the number of matched points used in the optimisation is shown to decrease error and variance of the parameters. Error for using fewer than 35 points is not shown due to frequent non-convergence.

The performance of the point-feature calibration algorithm is predicated on the availability, and robustness of point features in the synthetic laser image. Since we have an abundance of data, we can afford to set very conservative thresholds. This results in very few points being successfully matched per frame.

Typically, the SIFT algorithm extracts 2000 key-points per image, the feature association phase of Equation (5.9) finds 200 potential matches, and between 4 and 10 of those lie within the pixel distance threshold specified in Equation (5.10). The error curves of Figures 5.12 and 5.11 therefore indicate that at least 20 images are required to produce a reliable calibration estimate. Figure 5.9 shows an example of an image pair where features have been matched.

The histograms in Figure 5.13 represent the distribution of reprojection error for the matched points before and after optimisation. The effect of the optimisation is that mean reprojection error has minimised and shifted the histogram left towards



Uncalibrated Calibrated

Figure 5.13: Comparing histograms of reprojection error of an uncalibrated (left) and calibrated (right) system. The distribution on the left is measured using an initial incorrect calibration of $\mathbf{g}_0$. After optimisation, a distinct peak appears in the distribution.

the y-axis. This demonstrates that the majority of the laser points have moved closer to their image counterparts, suggesting that the matches are indeed correct and influential. The tail end of the histogram has reveals few outliers.

5.2.3 Discussion

This section proposes a method of calibration that automatically identifies and matches point features between images of differing modalities. Using these matches we formulate the camera–laser calibration based on the minimisation of the reprojection error. We show that a dense synthetic laser image offers a sufficiently close approximation of an equivalent camera image to allow at least partially successful feature matching. To account for the very low rate of positive feature matches, multiple images are used to obtain a sufficient number for reliable calibration.

The generation of the dense laser image is made possible by the slow travelling speeds of the *NABU 3* and the GEM (less than 5mph). On the RobotCar, viable points are only discovered during periods of slow travel speed, therefore increasing

the processing requirements; it is common for there to be no feature matches in an image. It is not clear what the effect of a low match *rate* are — as opposed to low point *count* which adversely affects calibration (Figure 5.12).

There are two reasons why this particular calibration method may not extend to 3D laser scanners. The first is that laser scanners such as those produced by Velodyne produce a sparse point cloud. This sparsity translates to sparsity in image space, thus not fulfilling the geometric requirements of visual feature extractors.

The second reason is the variability between individual lasers intensity and frequency response within a scanning unit; each lasers response curve would require calibration with the others to produce consistent appearance.

5.3 Conclusions and future work

This chapter was about the automatic calibration of a single camera–laser pair. We use 'automatic' here to mean 'without user input or intervention', as early methods discussed in the literature survey typically involve either constructed calibration targets or manual data association.

The contributions of this chapter are twofold; to present an analysis of camera–laser registration formulated using Normalised Information Distance, and to propose a new method of automatic calibration that matches point features discovered in both observations.

It is significant that all tests were carried out on a camera and 2D laser scanner that do not have overlapping fields of view. This configuration would normally require an intermediary sensor in order so solve the calibration, such as the PTI motion tracker discussed in Section 4.7.3. The visual odometry–supplied ego-motion, being locally very accurate [18], allowed eventual overlap of observations and hence no intermediary sensor or reference frame was required. Accurate odometry, and visual odometry systems in particular, are ubiquitous on mobile robots therefore we consider

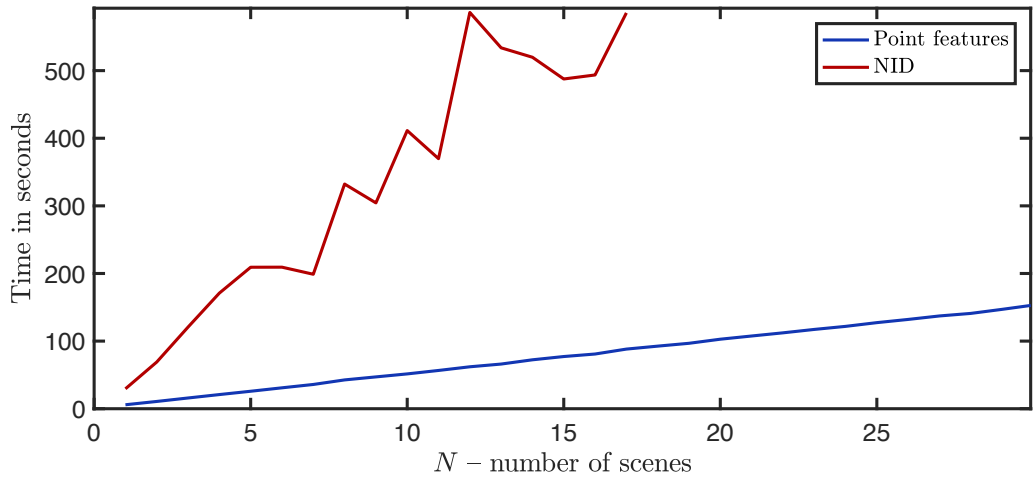


Figure 5.14: Computation time for NID and point-feature calibration algorithms for increasing number of scenes. The effects of the Nelder-Mead search and the manipulation of millions of points is apparent in the NID computation time. The linear increasing time of the point-features algorithm is due to the expensive feature extraction and matching sequence; the optimisation itself is negligible.

it reasonable to use it for benchmarking.

In terms over computational complexity, the NID objective function's expense is due to large numbers of points that are manipulated (in the order of 400000 per frame). The point-feature method involves extraction and matching of features (2.8 seconds per frame), requiring an initial dense synthetic laser image (5 seconds per frame). The subsequent least squares problem is solved appreciably fast due to efficiencies of vector programming and relatively small match pool. Both implementations were written in Matlab and can be further optimised and parallelised. Figure 5.14 shows the relative cost of each method with respect to the number of frames to be processed.

We have not factored the vehicle's speed into the analysis of our results but it is worth discussing. It is likely that speed of the vehicle will affect calibration, possibly adversely. The camera image tends to blur at higher speeds and in lower light, creating indistinct features and ambiguity in feature location. The point cloud for constant length will become more sparse, reducing the number of observations in a given frame. We must also consider that the visual odometry system may also

affected, particularly in low light. Future work should consider the factor of speed when approaching data-driven calibration.

5.3.1 NID calibration

There are three methods in the literature we have covered that offer near-true automatic calibration, short of supplying a rough initial calibration estimate [41, 52, 61]. Common between these approaches is the eschewing of explicit feature correspondences in favour of featureless similarity metrics, removing the susceptibility of the registration to false-positive feature matches. Each of the methods show promise in the pursuit of truly automatic calibration.

Of the global metrics referenced above, NID favourably offers calibration that does not depend on the geometric structure of the 3D point cloud. We investigate the effect data preprocessing has on calibration. Laser intensity as reported by the laser scanner has too low a dynamic range to be compared directly to the distribution of camera intensity. Histogram equalisation of the input data solves this problem directly, and has a significant positive effect on calibration. Gaussian smoothing can expand local features, although mutual information is highly robust to noise and smoothing should not be required for noisy sensor readings.

Fusing multiple scenes is shown to greatly improve the robustness of calibration and smooth the basin of convergence. The abundance of real-world data means that simply adding more scenes is a tenable solution.

The use of a featureless registration metric is attractive. There is consensus however that performance varies significantly depending on the appearance of the scene [52, 61, 88], which can even lead to failed calibration. Levinson and Thrun [41] report that their calibration algorithm succeeds despite the presence of poor scenes.

Our findings are largely in-line with those of other authors; fusing multiple scenes generally does improve robustness. We digress in that the indiscriminate fusing of multiple 'failure' cases still results in calibration failure and thus intervention is re-

quired. This point makes for an apt transition into Chapter 6 where we analyse these failure cases and propose a method that actively selects scenes according to their individual contribution to the optimisation problem.

An added value to using an information theoretic measure is that it is not limited to using only the luminance values of both the camera and laser. Luminance is a natural choice since both sensors' readings are dependent on the reflective characteristics of a surface. In theory, any two modalities that are statistically related could be used.

Zhao et al [96] experiment with different combinations of 'attributes' derived from the respective sensor's data, and measure how each affects calibration performance. Their results are mixed, with some combinations offering no clear evidence that the data is mutually dependent.

Further experimentation of these combinations on the ORI data would prove useful, particularly if applied specifically to 2D laser scanners. For example, if it is assumed that motion is planar, the angle subtended between consecutive planar laser points could prove to be a unique identifier of a surface in regularly structured urban scenes.

The NID formulation as presented in this chapter can be improved. We have not investigated the causes of high frequency noise present in the objective function. Future implementation may benefit from more robust formulations of the joint distribution, which may include B-Spline approximation [23] and spatial weighting [20].

We would submit that future improvement lies predominantly in augmenting the NID calibration method with techniques to identify scenes and regions within those scenes that offer a stronger signal. This is discussed at greater length in Chapter 6.

5.3.2 Auto point feature calibration

In Section 5.2 we present and demonstrate our method of matching point features common across camera and synthetic laser images using a commonly used feature detector [46]. The result of a positive match is a hard constraint between the 2D camera image coordinate and the 3D laser point (found by nearest neighbour to the laser match in $\mathbb{R}^2$). As expected, the rate of positive feature matches is poor but since we are afforded a virtually infinite number of scenes to extract feature matches, opting for a high-precision and low-recall is viable.

We found the calibration to be accurate and consistent. The auto-feature method produced results on both the GEM and RobotCar comparable to that of NID, and contrarily strong performance on the *NABU 3* using the warehouse dataset.

Our implementation of the SIFT feature extractor has not been optimised for the use with laser-generated images. Optimisation of the SIFT parameters and adjustments to the matching algorithm could potentially increase positive matches between a factor of 4 and 22 [75].

The outlier rejection we propose (pixel distance threshold), while functional, is rudimentary. That is to say it would not be robust to outliers should the rejection threshold need to be increased.

More experimentation is required to determine how robust the feature extraction is to the laser density. Should the vehicles be travelling faster, or a fundamentally sparse laser scanner (e.g. Velodyne) be used, could we still reliably extract mutually visible features?

In keeping with a policy of continual and life-long calibration, a method that continually selects only strong matches to maintain an accurate calibration over time, would also be beneficial.

Chapter 6

Intelligent Scene Selection

In Chapter 5 we demonstrated calibration of a camera and laser scanner using natural scenes as the robot would ordinarily observe them. This was accomplished in part by using the information theoretic measure *normalised information distance*. While this measure was found to be robust to a number of visual artefacts such as noise and lighting conditions, we reported that some scenes cause the optimisation to diverge: calibration fails. This observation is consistent with other researchers who propose methods that use a global cost function [52, 61, 88, 95]. In each case, calibration robustness is somewhat improved by increasing the number of observations. In this chapter, we demonstrate that particular scenes result in failed registration, as exemplified in Figure 6.1, and argue that such observations should be excluded.

Calibration targets are designed to provide regular, strongly discernible signals in both the camera image and laser modalities which we are not afforded in a normal working environment. The cost surface of the 'good' scene in Figure 6.1 is thus defined and, while not technically *convex*, is *quasiconvex*. This particular shape is characteristic of information measures used for image registration, which we demonstrate in Section 4.5. For the purposes of this chapter, the distinction between *convex* and *quasiconvex* is not significant for optimisation since both are smooth and manifest

unique minima. Any further mention of 'convexity' will simple mean 'quasiconvexity', or a function that satisfies $f(x_i) \leq \max(f(x_1), f(x_2)) \quad \forall x_1, x_2 \ : \ x_i \in [x_1, x_2]$.

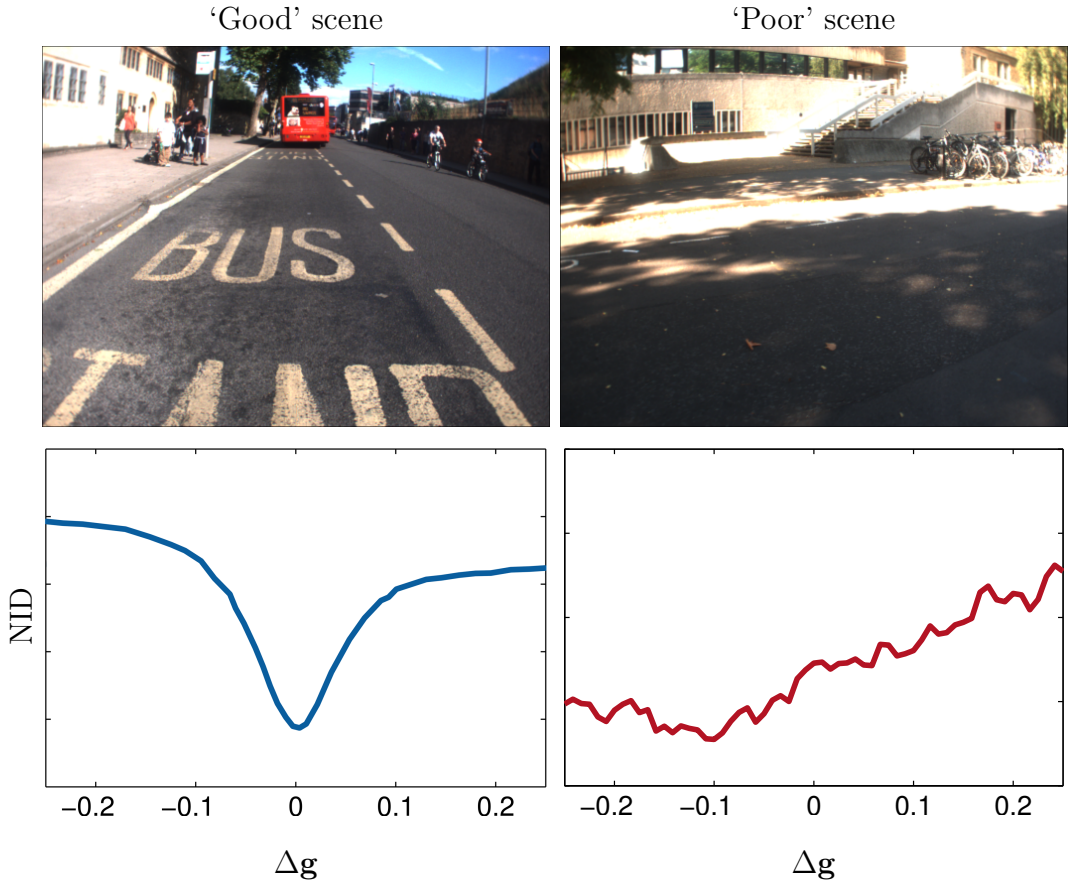


Figure 6.1: Scene variation and cost-surface convexity: artefacts seen in the camera image such as shadows, over/under exposure can lead to extreme non-smoothness. An image not limited by the camera's dynamic range, and has contrasting geometry produces a smooth convex cost function (left).

This thesis considers the calibration of a 2D laser scanner and a camera, which to our knowledge has not been solved using a global cost formulation. We generate 3D swathes using the known ego-motion to simulate sensor FoV overlap, thus laser range error is composed with that of the trajectory. We also expect the resulting point cloud to be of lower metric accuracy than that of a similar fidelity 360° scanner due to errors in the estimation of ego-motion, and therefore provide less reliable localisation and calibration. A solution is to be selective in the data used in the optimisation, as it is evidenced in Chapter 5 that successful calibration is possible even with a more

unusual camera–laser configuration in particular circumstances.

In this chapter we propose a policy of *scene selection*, which selects scenes based on their individual contribution to the optimisation. The NID objective function is underpinned by the appearance and structure of the scene being observed. Our approach is to identify scenes that best constrain the parameters of the optimisation since there exist certain conditions that demonstrably cause divergence.

This first part of this chapter analyses and introduces the problem. We discuss the probable causes of failed calibration, comparing scenes that have found to induce convergence with those that have not.

We then propose a method that analyses the shape of the cost surface around the located minimum to ascertain the nature of convergence. As a measure of *smoothness* and *distinctness* of a converged optimisation, we define a 'convergence score'. Without proir knowledge of the scene appearance, or sensor configuration, we can successfully select scenes that are shown to reliably produce a more accurate and precise calibration.

The work in this chapter has been published and presented at The International Conference of Robotics and Automation in 2016 [72].

6.1 Effects of scene variation

A core premise of the information theoretic method of calibration discussed in Chapter 5 is that there are characteristics and features of the environment that manifest in both camera and laser observations. In our particular implementation, we rely on the measured pixel luminance and laser reflectance being mutually dependent on the surface properties of the observed object.

The assumption of mutual dependence between the two modalities we use holds generally due to the wavelength of the infra-red (IR) light used by lidar scanners

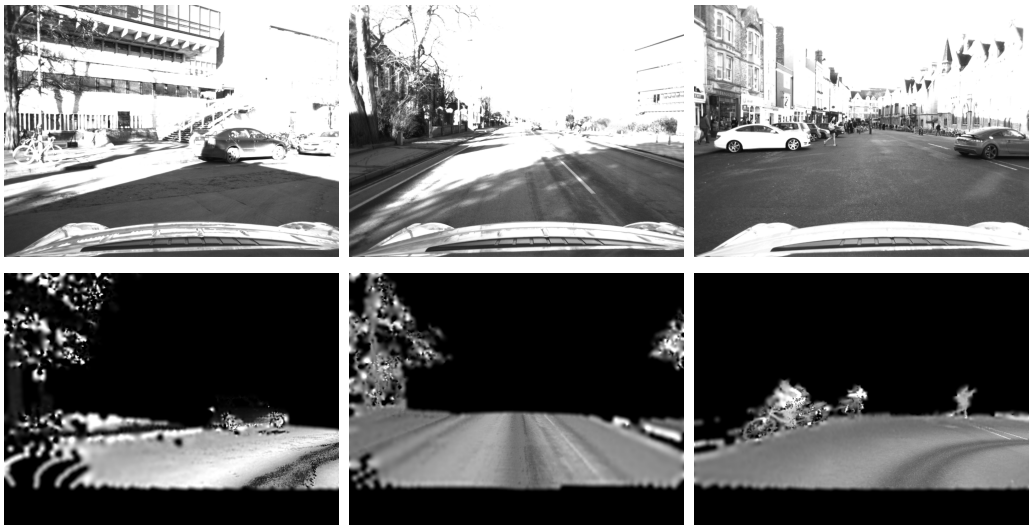


Figure 6.2: Camera image and corresponding anomalous laser image. The top row shows examples of scenes whereby the camera captures artefacts that laser (bottom row) cannot. For example, each camera image has varying levels of shadow and overexposure due to the poor dynamic range of the camera. We also see in the third column that moving objects have changed the scene between captures; the car and the pedestrian have moved between captures. These factors coupled with a scene that lacks strong geometric features leads to a high likelihood of divergence when optimising.

(905nm) being close to visible light[1] measured by the camera (350-700nm), and that visible and infra-red light are reflected proportionally by the object.

An exception to this assumption would be vegetation, which reports a comparatively higher laser reflectance than pixel luminance [35]. However, it is not necessary for lightness measured by the camera image to be proportional to lightness of the same object measured by the laser scanner. Recapitulating the point made in Section 5.1, differing modalities are adequately handled by the NID, which only concerns the *distribution* of measurements and not the *values* themselves.

There are conditions that are too extreme for even the NID to handle adequately. These are the cases which we consider in this chapter. Figure 6.2 shows examples of scenes whose appearance differs significantly between the laser and the camera. The

[1]Charge-coupled device (CCD) sensors are capable of measuring infra-red light directly but the lenses used often filter out IR light. We consider the case where only the visible light spectrum is observed

camera images are corrupted by over/under exposure, strong shadows and moving objects. It is in scenarios such as these which we argue calibration is likely to fail.

6.2 Scene selection via cost-surface analysis

Our foray into *intelligent scene selection* begins by probing the optimisation itself, ignoring the actual observation data for the time being. This section explores how one may assert that a minimum is potentially global, thus reflecting the true solution.

6.2.1 Verifying a potential minimum

The non-smooth optimisation problem in the case of calibration presents itself with multiple local minima; finding the global minimum in such problems is difficult. Thus, it is necessary employ measures to determine the likelihood that the found minimum is a plausible global solution.

In a maximum likelihood sense, the inverse-covariance of a Gaussian random variable is equal to the Hessian at the solution. It ensues that the Hessian is positive-definite, indicating that solution is indeed a minimum. We can apply this estimation of covariance analogously to the optimisation of NID. The Hessian of the cost function at the solution will not represent the inverse-covariance in a mathematical sense but the positive-definite requirement, indicating the solution is indeed a minimum, will still be valid. The experimentation in Section 5.1 reveals that the NID cost function has high-frequency noise causing local non-convexity. The estimation of the Hessian numerically would require a careful selection of finite differencing as not to differentiate the noise [37].

Due to the non-smooth nature of the NID cost function, we posit that the Hessian does not inform as to whether the found minimum is global as it measures only local convexity. In the following subsections, we consider the broad shape of the cost surface about the minimum to provide a better judgement of successful optimisation.

6.2.2 Global minimum via SVM classification

If we inspect the cost basin of the NID cost function around a global minimum, we notice that it takes a distinct shape. This is the case when using a synthesised scenario such as that illustrated in Figure 4.10, and in calibration using real data (example cost surfaces for NID can be found in Figures 5.6 and 6.1).

Stewart [76] considers the localisation of a camera with respect to a prior coloured 3D point cloud. The approach is formulated similarly to the calibration presented in this thesis, but the appearance of the laser points is obtained from a colour camera during the mapping phase, rather than the laser intensity values we use for calibration.

In order to determine whether a localisation has been successful, the author proposes a method that uses a support vector machine (SVM) to classify a cost-shape, pertaining to localisation, as a 'success' or 'failure'. We assess the feasibility of this method applied to calibration, where we only consider 'success' cases as suitable. Here we describe the method used to classify a scene as either being 'useful' or 'not useful' for calibration.

Generating scene descriptors from the cost surface

Recall from Section 5.1 the formulation of the cost function for the pairwise calibration of a camera and laser scanner using scene s as

$$D^s(\mathbf{g}_{cl}) = 1 - \frac{I({}^c\mathcal{I}^s; {}^l\mathcal{I}^s)}{H({}^c\mathcal{I}^s, {}^l\mathcal{I}^s)}. \tag{6.1}$$

The cost function in Equation (6.1) is then used to generate the scene descriptors as follows. For each scene, the camera–laser transform is solved, about which a cost surface is generated. We use the uniform partition for each of the 6-DoF parameters (denoted $\alpha = 1 \dots 6$) $\{\mathbf{g}_0^\alpha, \dots, \hat{\mathbf{g}}, \dots, \mathbf{g}_n^\alpha\} \subset [\hat{\mathbf{g}} - \Delta\mathbf{g}^\alpha,\ \hat{\mathbf{g}} + \Delta\mathbf{g}^\alpha]$ of $n = 9$ to generate independent discrete cost surfaces. The value for n was chosen to minimise the computational overhead while ensuring that that the characteristic shape of the objective

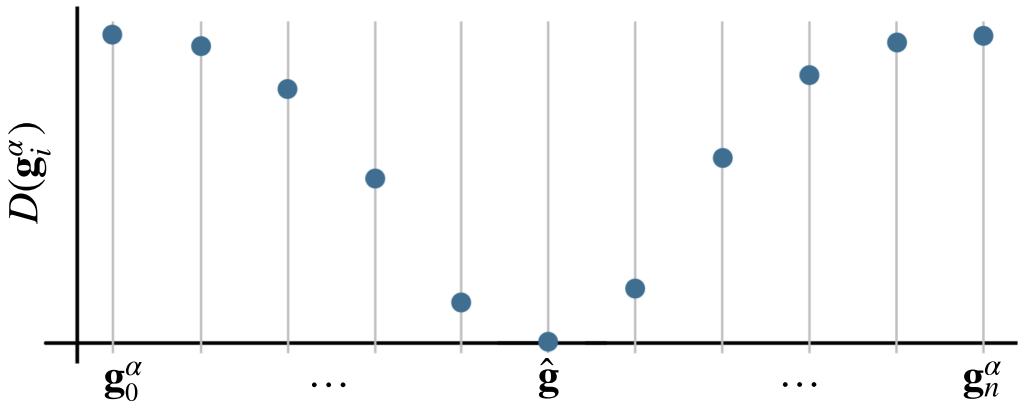


Figure 6.3: Creating a scene descriptor by sampling the cost function. The scene descriptor Φ_α for a given parameter α is formed by creating a partition $(\{\mathbf{g}_0^\alpha, \ldots, \hat{\mathbf{g}}, \ldots, g_n^\alpha\})$ around the located calibration $\hat{\mathbf{g}}$, and finding the corresponding values according to Equation (6.2).

curve is captured adequately. We choose a $\Delta\mathbf{g}$ of 0.2 to sufficiently capture the basin of convergence. The cost surface for each α is normalised to $[0, 1]$ to distil only its shape. The feature for α can be described by the tuple

$$\Phi_\alpha = (D^s(\mathbf{g}_0^\alpha), \ \ldots, \ D^s(\mathbf{g}_n^\alpha)) \,. \tag{6.2}$$

Figure 6.3 shows a visualisation of this descriptor of a single dimension. The final descriptor can then be formulated as the concatenation of these individual features (one for each DoF), with the minimum cost appended

$$\mathbf{\Phi} = (\Phi_1, \ \ldots, \ \Phi_6, \ D^s(\hat{\mathbf{g}})). \tag{6.3}$$

The resulting dimensionality of this descriptor is $6n + 1$. A visualisation of the descriptors for a labelled dataset is presented in Figure 6.4 whereby the positive examples are highlighted.

Creating labelled datasets

A supervised learning approaches requires a fully labelled dataset, an endeavour not without its own challenges and shortcomings. We do this by first selecting a large

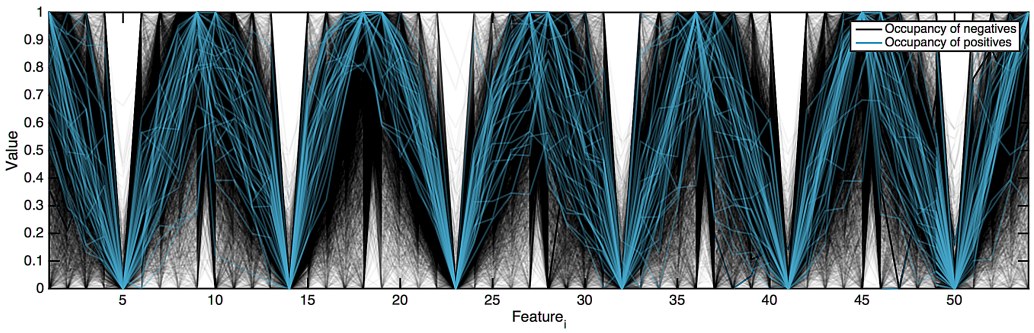


Figure 6.4: Scene descriptors formed from the cost function. Presented here is a visualisation of a labelled set of descriptors; a set of 5000 descriptors overlain in a single graph. Each descriptor represents a concatenation of discrete approximations of the cost basin in a given 6-DoF parameter. The blue 'positive' descriptors are seen to occupy a distinct part of the feature space.

vehicle dataset, and identifying *valid* scenes.

There are two conditions that must be met for a scene to be labelled as valid. The first is a requirement that the calibration achieved using only a single query scene falls within a predetermined error range; scenes causing calibration error to fall outside this range can be rejected as we assume the cost function is divergent. Those scenes falling within the allowable error bounds undergo manual inspect to determine if the resulting cost surface around the found minimum is convex and smooth. This method of labelling is largely subjective since there is no impartial and reliable metric of scene efficacy.

Training and evaluation

The labelled dataset is then split randomly into a training and testing set in a 4:1 ratio respectively. Using a radial basis function, the SVM is then trained with cross-validation.

The results of experimentation are shown in Figure 6.5 which compares the precision–recall of this SVM method to the 'total variation' method presented in the following section. It is worth noting some of the limitations of representing the performance using precision–recall as applied here. The first is that the dataset is labelled in a

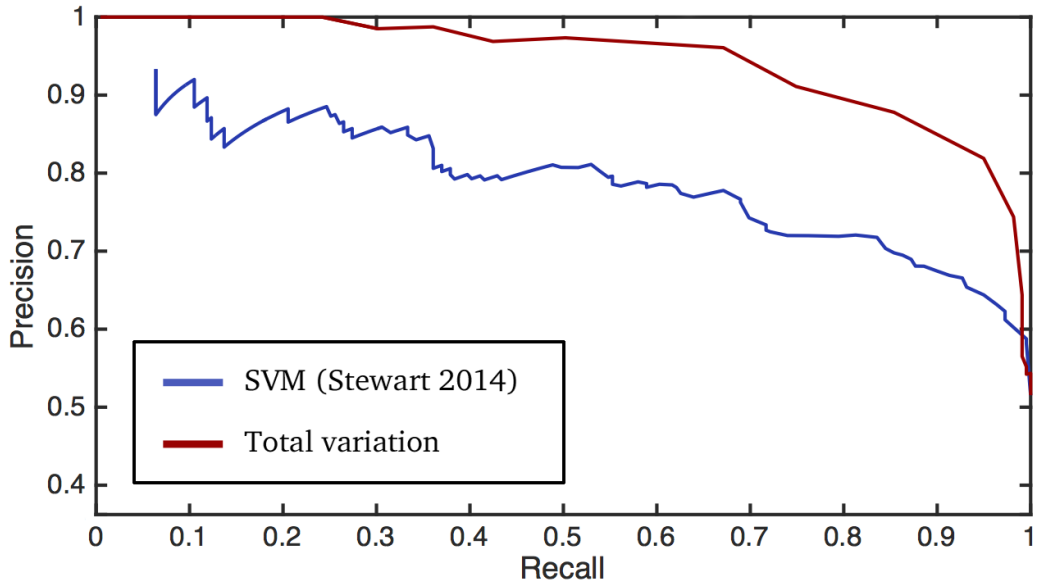


Figure 6.5: Precision–recall comparison of methods SVM and 'total variation' to classify scenes as either 'success' or 'failure'.

binary fashion, where in reality the is no sharp distinction between a 'good' and 'bad' scene. The second is that the labels are produced by an expert, albeit subjective, user based on a set of assumptions of what constitutes a 'good scene'.

The shape of the cost surface is implicit, and is learnt by the SVM. A significant limiting factor of this approach is the requirement of learning a model from labelled data, making it antithetical to true automatic calibration. For our purposes it is more appropriate to use a continuous scoring scheme that allows for selecting the most informative scenes and without having to learn a model, which leads us into the next section.

6.2.3 Scene scoring based on cost-basin shape

Learning a model based on labelled training data is a particularly untenable solution since this thesis is premised on the philosophy of end-to-end autonomy. An unsupervised approach would need to automatically determine which scenes produce successful calibration. We instead investigate ways to interpret the shape of the cost function which provide an indication of whether the observations will constrain the

calibration parameters, and allow for the discovery of 'good' scenes.

A second consideration is the assumption that 'efficacy' of a scene is not simply a binary categorisation. This section details the design of a relative measure of efficacy such that those scenes that contribute more reliably to the calibration can be selected.

The Hessian at the located minimum, if positive-definite, reveals that the optimiser has indeed reached a stationary point, yet it lacks insight to the likelihood of this solution being global. Through experimentation of synthetic and real data we discover that the shape of the NID cost function about the minimum has a certain characteristic. This finding is in concert with others that have studied information theoretic measures for image registration [20, 63, 76].

We propose an algorithm that identifies the characteristics of a global minimum as a relative measure of scene efficacy. We define two quantities that measure the quality of the optimisation; monotonicity or 'smoothness', and 'distinctness'. The former is a requirement for convexity, while the latter affords us a relative measure for cost surfaces which are already smooth. The combination of these two quantities is what we term the *scene score*.

Monotonicity via total variation

Unimodal optimisation requires that a cost function be convex[2]. We define a function to express this monotonicity as a proportion of change in cost ΔD^s to the total variation of the cost function.

The expected output of this function will be in the interval $[0, 1]$ where 0 indicates that there is no overall change in cost, while 1 indicates a purely monotonic function. For simplicity, each of the six parameters is treated independently.

As a function of two uniform partitions $\{g_n^-, ..., g_0^-\} \in [\hat{\mathbf{g}} - \Delta\mathbf{g},\ \hat{\mathbf{g}}]$ and $\{g_0^+, ..., g_n^+\} \in [\hat{\mathbf{g}},\ \hat{\mathbf{g}} + \Delta\mathbf{g}]$ split at the minimum $\hat{\mathbf{g}}$, the total variation is given by

[2]quasiconvexity and pseudoconvexity also presume a unique minimum

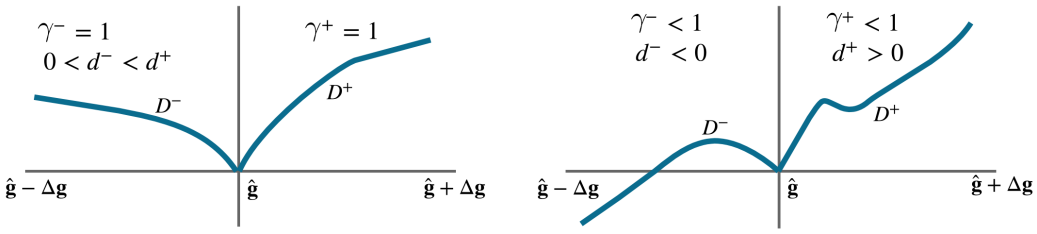


Figure 6.6: Illustrative cost surfaces explaining 'smoothness' and 'distinctness' measures for scene selection. The left hand cost surface demonstrates the 'ideal' case where the cost function is quasiconvex and therefore monotonic on both sides of the minimum $\hat{\mathbf{g}}$ ($\gamma^+ = \gamma^- = 1$) and is distinct ($d^+ > 0 \cap d^- > 0$). The right hand graph shows a sub-optimal minimum; it is not local ($d^- < 0$) and the function is not monotonic ($\gamma^+ < 1 \cap \gamma^- < 1$).

$$\begin{aligned} V^- &= \sum_{i=1}^{n} |D^s(g_i^-) - D^s(g_{i-1}^-)|, \\ V^+ &= \sum_{i=1}^{n} |D^s(g_i^+) - D^s(g_{i-1}^+)|. \end{aligned} \tag{6.4}$$

The superscripts $+, -$ indicate a positive or negative perturbation about the minimum respectively and will in future be abbreviated to $\pm$. We can thus define a relative measure of monotonicity by computing the fraction of variation to total variation as follows,

$$\begin{gathered} \gamma^{\pm} = \frac{|\Delta D^{s\pm}|}{V^{\pm}} \\ \text{where } \Delta D^{s\pm} = D^s(\hat{\mathbf{g}} \pm \Delta\mathbf{g}) - D^s(\hat{\mathbf{g}}). \end{gathered} \tag{6.5}$$

The interval $[0, 1]$ that γ maps to can be verified by noticing that any cost change has to be bound between the total variance $0 \leq |\Delta D(\mathbf{g})| \leq V$.

Distinctness of the minimum

To measure the 'distinctness' of the minimum, analogous to the variance of a least squares problem, we calculate the average gradient $\frac{\delta D(\mathbf{g})}{\delta \mathbf{g}}$ within the intervals listed above. We therefore define 'distinctness' as

$$d^{\pm} = \frac{D^{\pm}(\hat{\mathbf{g}} \pm \Delta\mathbf{g})}{\Delta\mathbf{g}}. \tag{6.6}$$

The results of these definitions are explained visually in Figure 6.6, which shows how γ and d vary for different possible cost function shapes. The score given to a particular parameter α can be computed by taking both these factors into account, and we define it as

$$\mathcal{K}_{\alpha} = \min(\gamma^{-}d^{-}, \gamma^{+}d^{+}). \tag{6.7}$$

The score computed in Equation (6.7) considers the effect of varying each of the 6-DoF parameters independently, thus $\Delta\mathbf{g}$ denotes the scalar perturbation in only a specific parameter. We therefore calculate the scene score $\mathcal{K}_{\alpha}$ for each parameter α, leaving us to define a *scene score* conservatively as simply the minimum of the six individual parameter scores

$$\bar{\mathcal{K}} = \min_{\alpha \in \{1 \ldots 6\}} \mathcal{K}_{\alpha}. \tag{6.8}$$

6.2.3.1 Results

We present the results of *scene scoring* methods detailed above, and begin by verifying that the algorithm has indeed behaved as intended; Figures 6.8 and 6.9 show example scenes that have scored the highest and lowest respectively. The cost functions of those high scoring scenes indeed exhibit mostly smooth and distinct cost surfaces, while those exhibiting a low score are non-convex and non-smooth.

The corresponding camera images of those figures confirm intuitions we had in Section 6.1, that extreme phenomena such as shadow, over/under exposure and lack of structure can reduce calibration quality.

It it worth reiterating the goal of the thesis of this chapter: to produce more

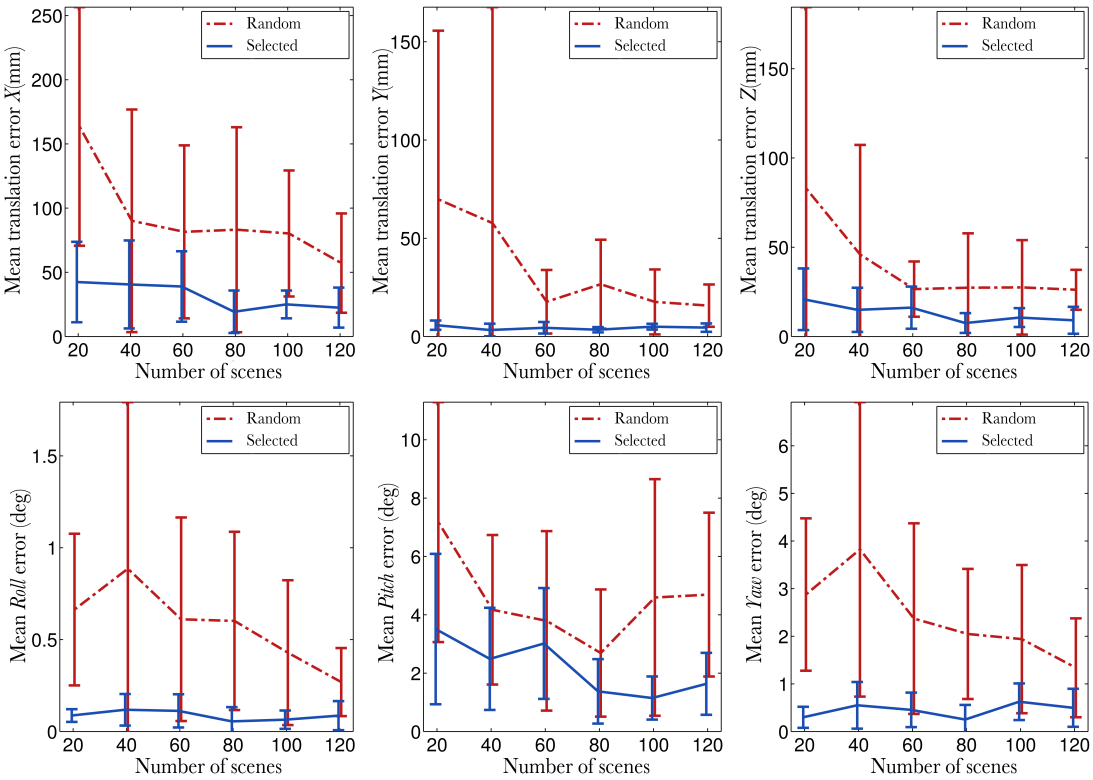


Figure 6.7: Mean squared error exhibited when using a randomly selected subset of scenes, and when using our 'total variation' scene selection method. Results are shown for each 6-DoF parameter individually. Error bars indicate $\pm 1\sigma$.

reliable and more accurate calibration. While we can speculate that these high-scoring scenes offer better calibration by virtue of their cost function and appearing alone, we offer quantitative calibration results to reaffirm the above results. As in Section 5.1, we repeat the experiment of evaluating a calibration using varying number of scenes. This experiment instead uses the top N scoring scenes. The results, shown in Figure 6.7, indicate a clear improvement in both accuracy and variance of calibration where even using a few scenes outperforms a large random subset. The error shown is calculated using the difference between the inferred stereo camera–camera transform (obtained from Section 4.7.1) and the transform supplied by the manufacturer.

This result is also reiterated in Figure 6.10 which shows the distribution of calibration solutions of 100 individual scenes, taken from a GEM dataset, for both the *intelligently* and *randomly* selected scenes. In this case, the camera–laser 6-DoF cali-

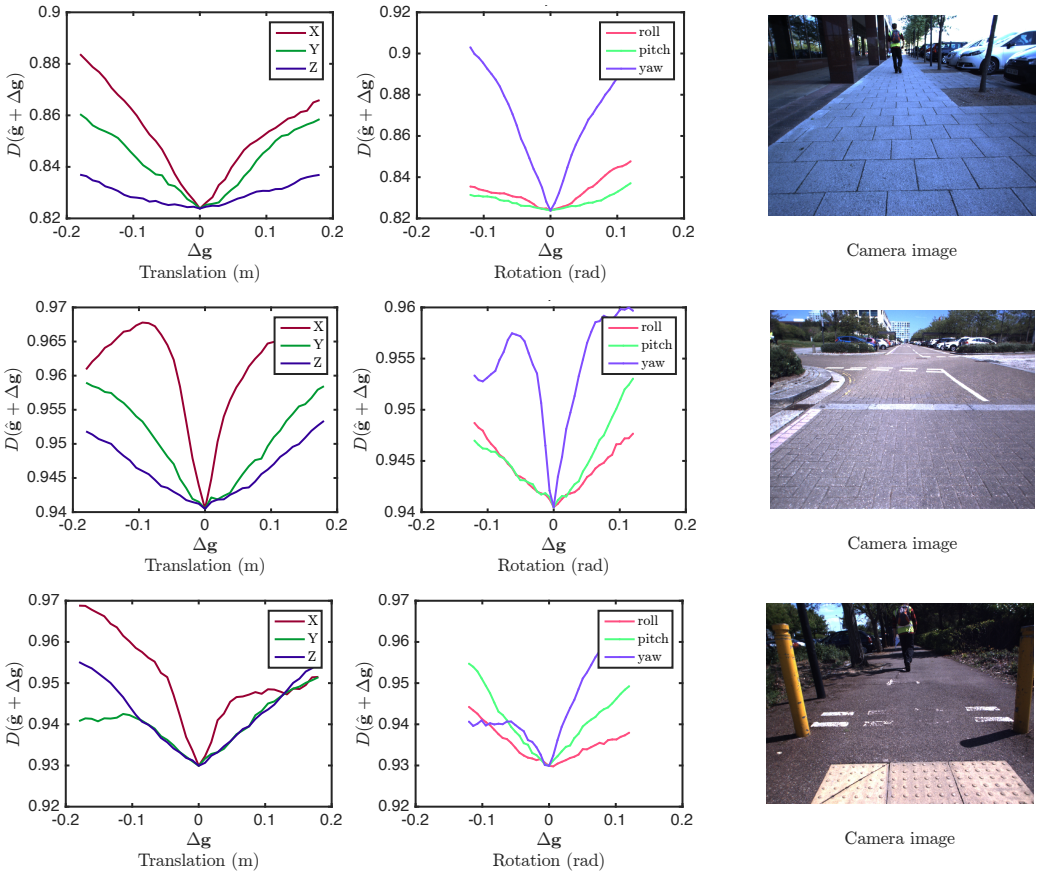


Figure 6.8: Three high-scoring scenes selected using the 'total variation' method. Each row corresponds to a particular scene. The first and second columns report the cost function around the minimum by perturbing translation and rotation elements respectively. In each case, the cost function is convex and relatively smooth.

bration is presented. The ground truth values obtained using the motion tracker can be found in Table 5.1, and are also indicated by the horizontal line.

Again, the result is clear; the solutions of *intelligently* selected scenes are more closely distributed around the ground truth, while randomly selected scenes often exhibit divergence.

6.3 Concluding remarks

In Chapter 5 we formulated a method of calibration that measures the overlap of information contained in both the camera and laser data. The *normalised information*

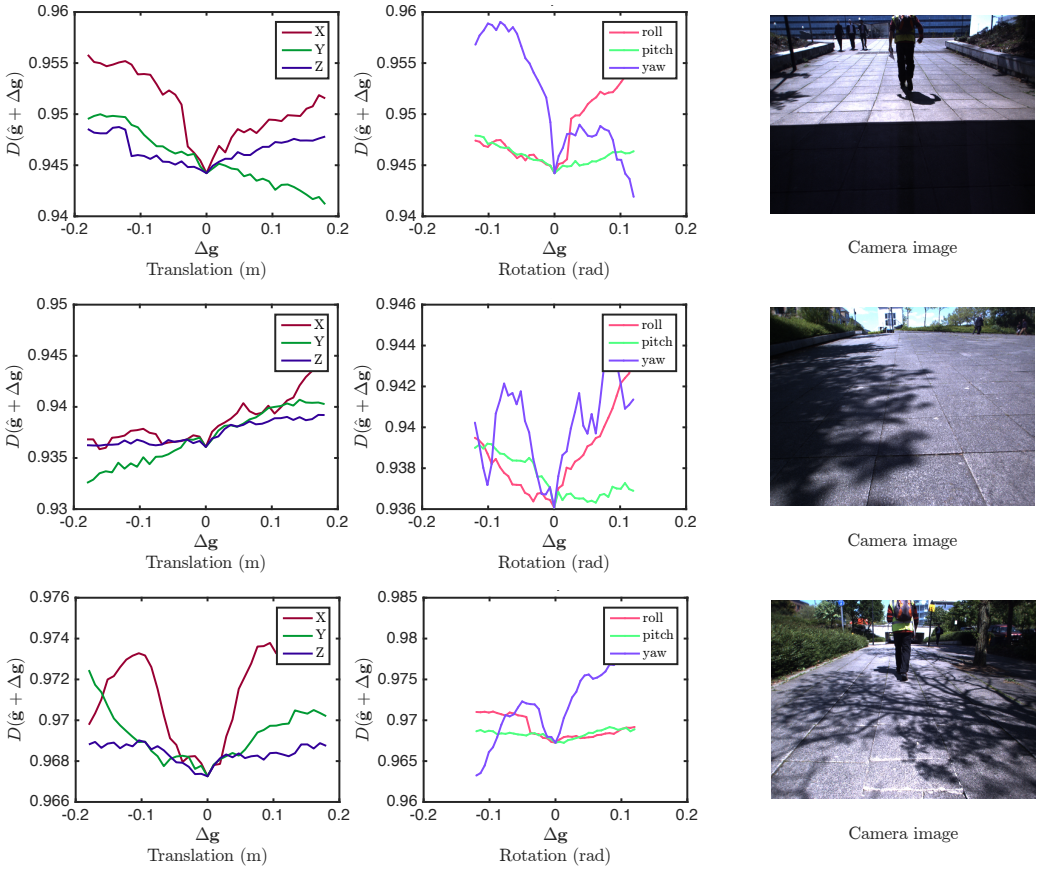


Figure 6.9: Three low-scoring scenes selected using the 'total variation' method. Each row corresponds to a particular scene. The first and second columns report the cost function around the minimum by perturbing translation and rotation elements respectively. The cost functions are show to be non-convex, and in some cases the minimum is non-global.

distance is particularly suitable since it allows sensors using differing modalities or response curves, and for using real-world observation. We discovered however that this calibration method is not robust and is heavily dependent on what is being observed. While some authors assert that adding more observations overcomes this issue, we have demonstrated that this is not the case. With reference to Figure 6.7, adding observations does indeed improve calibration accuracy and reliability, but being selective about which observations are used is always better. Figure 6.11 makes the same case as it shows the distribution of scores, whereby the majority of scenes have a negative score which indicates that the cost function is non-convex or that the computed minimum is non-global. We also gain insight into the poor

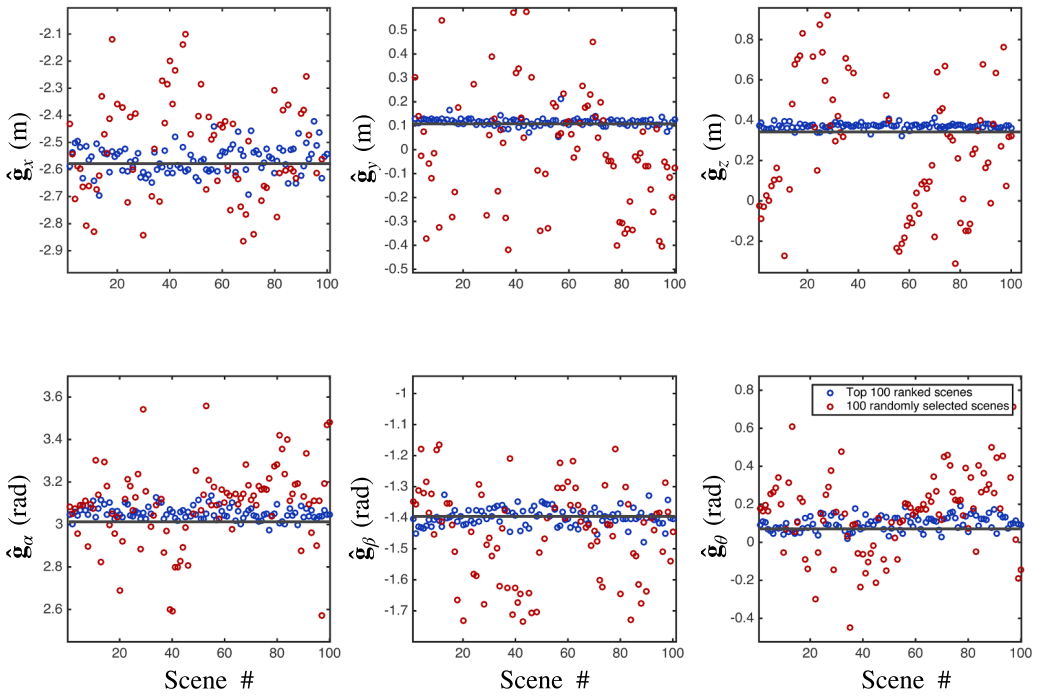


Figure 6.10: Plot of solutions using *random* vs. *selected* scenes using the 'total variation' method. Each graph indicates the estimated value of each of the six parameters in $\mathbf{g}_{cl}$. The distribution is significantly tighter when using highly scored scenes, while using random scenes does not indicate any evidence of convergence. Scenes were selected from a Milton Keynes dataset using the GEM. The horizontal line indicates ground truth (see Section 4.7.3).

calibration result of the NABU using the warehouse dataset. The distribution of scores is much narrower, as scene variation in the warehouse is lower compared to an urban environment. There are far fewer high-scoring scenes relative the 10km Oxford dataset.

This chapter investigated way in which these observations can be selected *intelligently*. We propose a method that takes advantage of the optimisation itself by noticing that a 'global minimum' of the NID cost function is distinctly convex and smooth within a certain distance about that minimum. This method has the advantage of being agnostic about the observation data and does not require the explicit extraction of features. Instead we rely on the assumption that important features manifest themselves in the cost function. While the SVM classification of a minimum

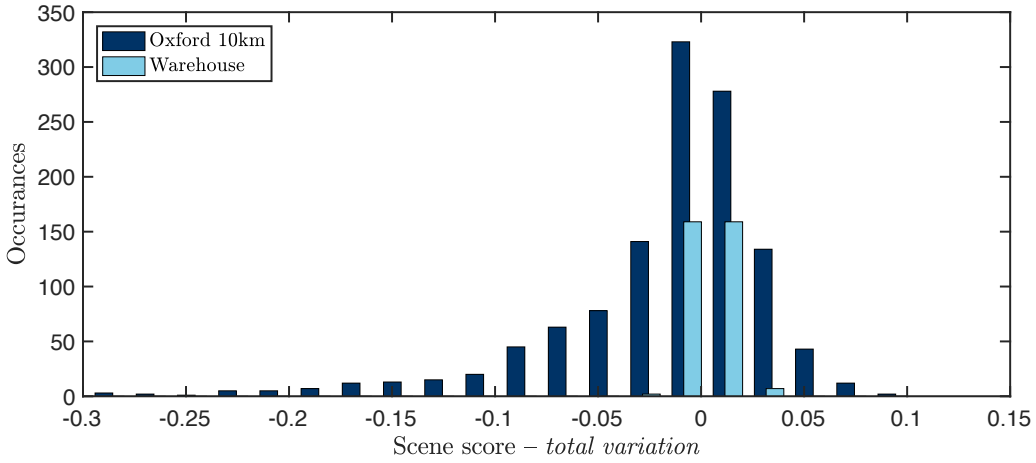


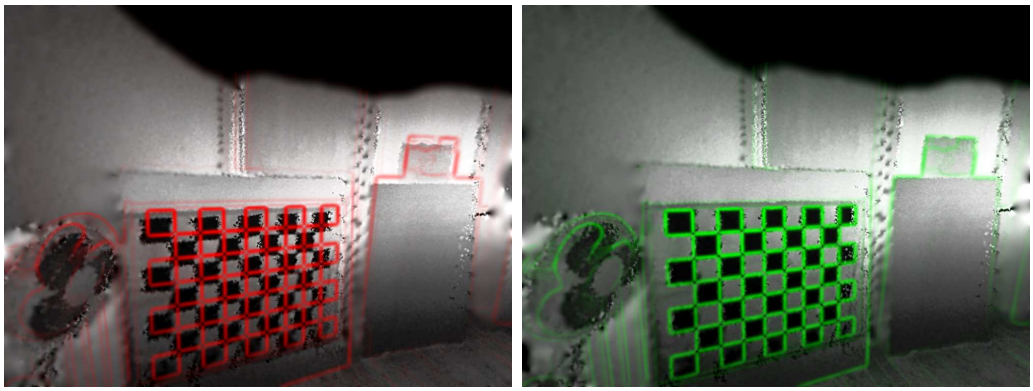
Figure 6.11: Distribution of scene scores for an Oxford 10km and Warehouse dataset. The histogram shows that the majority of scenes score below zero on the Oxford dataset, indicating a non-convex objective function. We strive to select the few scenes on the higher end of the scale. The distribution of warehouse scores is much narrower with far fewer high-scoring scenes.

was shown to be successful, acknowledge the problem of providing meaningfully labelled training sets. A simple binary label and classification does not allow us to find the 'best' scenes and leads to polluting a positive training set with negative labels and vice-versa.

Overall, the proposed scene scoring method has shown to provide a reliable measure on whether a particular cost surface can be characterised as 'smooth and distinct'. These results are corroborated with improved precision and accuracy exhibited in Figures 6.7 and 6.10.

6.4 Future work

In this chapter we consider the case of calibration which is purely data driven. There are environments which potentially do not exhibit the richness of features that can aid calibration. It should then be possible to supply fixed targets in a robot environment (or leverage existing ones) to aid calibration. An example of such a scenario is offered in Figure 6.12, whereby a short traversal of a hallway included a large



Uncalibrated Calibrated

Figure 6.12: Adding a checkerboard to a bare scene. A visualization of an office hallway scene with uncalibrated (left) and calibrated (right) sensors. The proposed scene selection algorithm highlighted this scene as a 'good' candidate.

checkerboard. The scene selection algorithm highlighted this scene as suitable for calibration, suggesting calibration can be assisted by strategically positioned targets, without altering the algorithm.

We acknowledge that there are drawbacks to our proposed approach. The cost function has to be sampled numerous times (between 60 and 120 depending on chosen resolution) and inspection of only the cost surface offers no insight to the nature of the scene that contributes to successful calibration. We are left only with intuition. The following section offers preliminary research in using image-based features to determine scene efficacy with regards to calibration.

6.4.1 Scene selection based on image features

This section describes how we attempted to score scenes based on the features that are present in the camera image. The types of features that manifest in the scene that contribute to a successful calibration (or localisation) are not obvious. From the experimentation in Chapter 5 we notice that certainly scenes with greater structure, uniform textures and strong edges produce more reliable calibration, yet the challenge is to describe these features in a mathematically meaningful way. We therefore

explore how the presence of commonly used image features may provide insight to the confidence with which we can calibrate using a particular scene. We consider two different scene descriptors, which are used as the input to an SVM trained using a radial basis function with cross-validation as in Section 6.2.2.

6.4.1.1 Bag of visual features

Visual features have been successfully used in the past to establish an image's likeness with another [19]. In particular, the combination of the type and location of certain features can uniquely describe an image. Using what is called a 'bag of words', a measure of the frequency of image features, Dequaire et al [22] present a method that predicts likelihood of a successful localisation in a visual teach-and-repeat system. Similarly, we attempt to classify scenes based on the presence of local image features.

Our goal is to obtain a representation of a camera image based on the presence of SIFT features [46]. As with the SVM classification method described in Section 6.2.2, we require a pre-labelled training and testing set and therefore use the same labelled dataset mentioned in that section. Again, this method also uses a binary label so we note the possible limitations; of the 'positive' scenes we cannot determine which are 'best'. For each scene we summarise the generation of a scene descriptor as follows:

1. Find all SIFT features in a set of images using the computer vision toolbox VLFeat [85].

2. Determine cluster centres of M features using k-means method. Each cluster centre represents a 'word'.

3. For each image in the set, determine which cluster or word each of the detected features belongs to.

4. Generate a histogram using the frequency of the presence of each word determined in 2. This histogram is the descriptor for the image.

For our experiments we use a 500-word ($M = 500$) representation for an image.

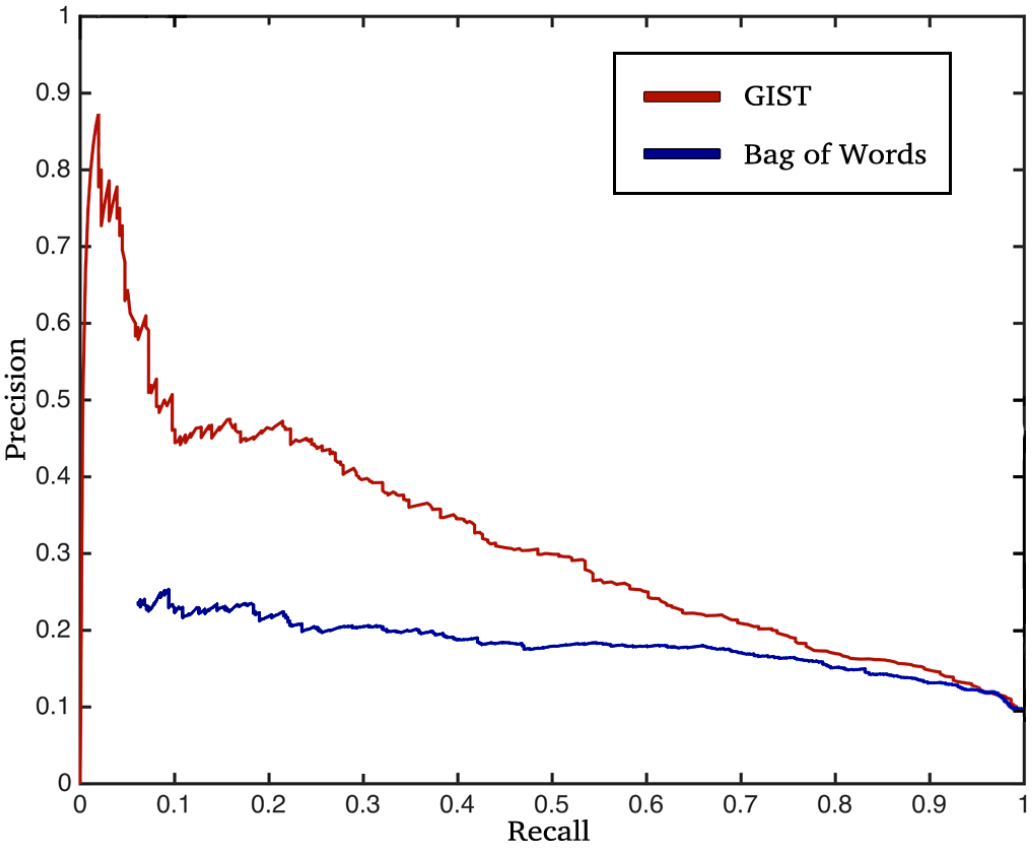


Figure 6.13: Precision–recall curves for scene classification using *Bag of Words* and GIST features. Scenes are classified as to whether the manifest cost function is convex with a distinct minimum (positive) or non-convex (negative).

6.4.1.2 The gist of a scene

Oliva and Torralba [58] have developed a descriptor that provide a 'holistic' representation of an image which is said to specifically capture properties of 'openness, naturalness, ruggedness and expansion'. The Matlab implementation we use for the implementation is available online [57]. As before, the descriptor generated from each image is used as the input to train the SVM.

6.4.2 Discussion on future work

Preliminary classification results are presented in the precision–recall curve in Figure 6.13. The learnt SVM models are evaluated in the same manner as in Section 6.2.2,

and using the same dataset.

We attempted to characterise a scene based on the visual features present in that scene. The two descriptors we used, GIST and *bag of words*, were shown to lack discriminatory power in this regard. The experimentation was however limited to training using 4000 examples in a 10:1 negative-to-positive ratio.

A more comprehensive test would need to be carried out to fully determine the efficacy of hand-crafted visual features. There may be a case to use a convolutional neural network to implicitly learn such features, but the burden of acquiring meaningfully labelled training data is prohibiting. Also not considered is the presence of geometric features in the 3D laser data which could offer evidence for a 'good' scene.

Simulated worlds offer robotics a wealth of data and testing scenarios. While the scene appearance can be accurately modeled in metric and colour space, for calibration the appearance would also need to be accurately modeled in infra-red (laser) space along with the variability and responses of the laser scanner. It is precisely the noise and uncertainty of multiple interacting components of a robotics system that make camera-laser calibration using scene appearance unreliable.

Chapter 7

Unified calibration by pose graph optimisation

In Chapter 5 we presented ways in which a camera–laser pair can be calibrated by using observations of the natural world. Robotic platforms are typically host to multiple cameras and laser scanners. This chapter therefore extends calibration to solve the joint calibration of multiple sensors by utilising the calibration of multiple sensor pairs, prior known calibrations and the resulting sensor network.

The data association problem becomes exponentially more difficult as the number of parameters to solve for increases. The multi-sensor calibration problem has been addressed in the past [26, 40, 59, 93]. Le and Ng [40] present a unified approach to extrinsic calibration and demonstrate that redundancy and shared constraints in the optimisation network significantly improves accuracy. Their method requires minimising a cost function according to known geometric constraints imposed by calibration targets.

Our proposed method couples suitably with entropy-based calibration presented in Chapter 5 and Chapter 6 in that it does not require explicit geometric constraints (other than pose constraints) between sensor pairs or groupings. Because the cali-

bration uses scenes from the robots environment we are not afforded the geometric constraints that are available when using known calibration targets; we proceed while being agnostic about the geometry of the scene around of the robot.

This chapter describes the method we use to jointly calibrate a system of sensors, where the pairwise registration is nominally an expensive computation. The graph optimisation we propose supposes a black-box pairwise calibration algorithm, and is thus impartial to the pairwise calibration process. We use alternating optimisation [11] to iteratively solve the parameters of each sensor–pair and then the whole system independently.

This alternating approach is particularly advantageous due to how the 3D point cloud are generated using a 2D laser scanner; the point cloud is dependent on the vehicle–laser relative pose, a variable which we are attempting to find. Recall the formulation of the point cloud projection equations in Section 4.3 which necessitate the point cloud's dependency on the calibration parameters. When calibrating cameras other than those which provide a pose source, we anticipate error due to the point cloud not being metrically accurate. By using an alternating optimisation we assume that an improved vehicle–laser calibration improves the camera(mono)–laser calibration and vice-versa.

The work in this chapter has been published and presented The International Conference of Intelligent Robots and Systems in 2015 [71] and The International Conference of Robotics and Automation in 2016 [72].

7.1 Method

The method presented here is a unified framework that treats each pairwise sensor registration as a black-box operation. The methods of [59] and [40] build pairwise constraints into the graph optimisation directly and, while it is theoretically possible to incorporate such constraints into our framework, it has not been covered here.

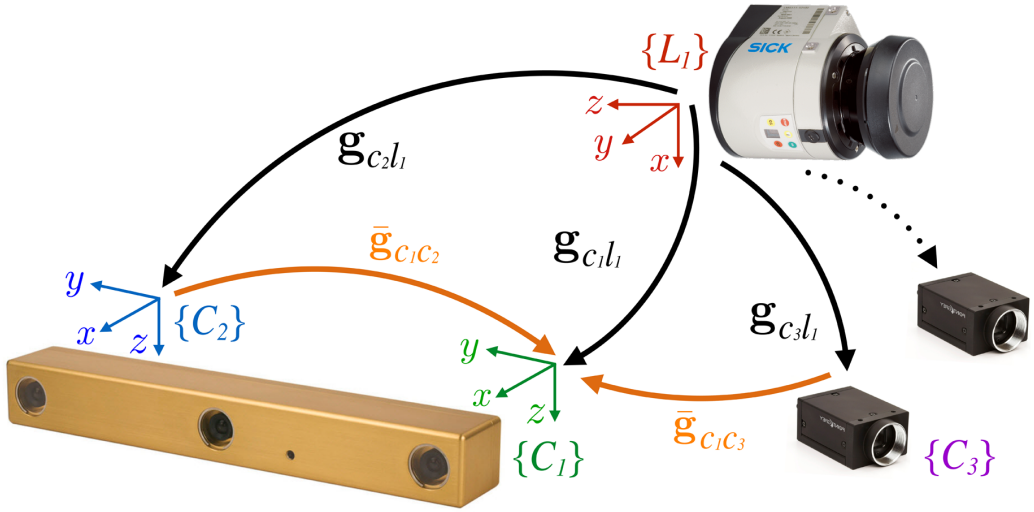


Figure 7.1: An example sensor network with notation indicated. The arrows indicate direction that transforms are expressed in, with the orange lines indicating the known constants. The black lines indicate that the relative pose must be solved for (lower-level).

Instead we separate the problem into two hierarchical levels, alleviating the need to solve expensive registration problems over multiples of the pose parameter space. The first, termed the *lower-level*, consists of a registration between two sensors, as covered in Chapter 5 for the laser–camera case. The second, or *upper-level*, combines the solutions found in the lower-level, with known camera–camera measurements (used to formulate the *constraints*) obtained previously by other means.

In this section we define a *constraint*, describe the formulation of the objective functions and how the upper- and lower-level are linked between consecutive iterations of the alternating optimisations.

We consider a typical vehicle setup such as that described in Chapter 3, which has cameras facing in different directions, a stereo camera mounted on the front of the vehicle and laser scanner mounted in a push-broom configuration. The schematic shown in Figure 7.1 depicts such a setup, indicating the notation for the reference frames and transforms that will be used in the equations to follow. The relative poses are denoted $\mathbf{g}_{AB}$ where entities expressed in the source frame B are registered in the destination frame A. The coordinate frame for the vehicle is denoted V and the

frames occupied by the cameras and laser scanners respectively are C and L.

In our formulation to follow, we initially assume that the vehicle is stationary, or that point clouds are generated using a 3D laser scanner, thus bypassing the vehicle ego-motion reference frame V (see Section 4.3). By modifying the error term of the simpler formulation we can then incorporate the vehicle frame and ego-motion, thereby allowing for a non-rigid point cloud as used in this thesis.

7.1.1 Notation

Throughout the chapter, we will use the standard form $\mathbf{g} \in \mathfrak{se}(3) \subset \mathbb{R}^6$ to represent a relative pose minimally, albeit with augmentations that cause significant distinction in meaning.

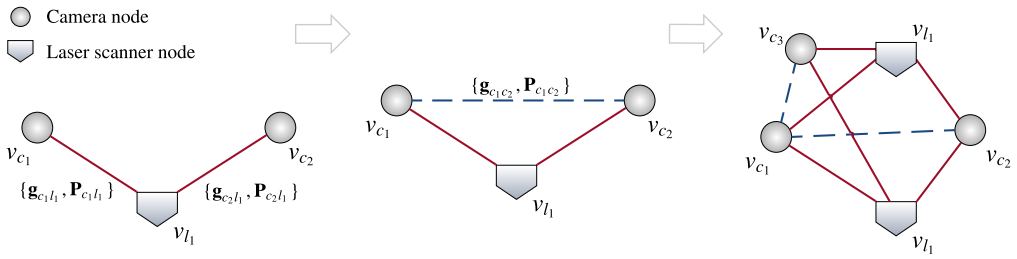
The *bar* variation is used when we introduce a constraint that is imposed by a measurement known outside of the calibration framework. For example, $\bar{\mathbf{g}}_{cc}$ is a measurement between two cameras obtained by some means unknown to the calibration system, and is constant throughout.

The *hat* symbol is used to distinguish a lower-level from an upper-level optimisation variable. The variable $\hat{\mathbf{g}}_{cl}$ is solved for in the lower-level, which involves the camera–laser registration process of Chapter 5.

The unmarked $\mathbf{g}_{cl}$ is a random variable that allows us to formulate upper-level pose constraints whose error distribution is normal, $\mathbf{e} = (\hat{\mathbf{g}}_{cl} \ominus \mathbf{g}_{cl}) \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_{cl})$. The $\hat{\mathbf{g}}_{cl}$ in this example is treated as constant.

7.1.2 Graph structure

The sensor network can be represented conveniently as a graph (Figure 7.3), described in terms of nodes (global transforms) and edges (rigid relative transforms). The graph is constructed with set of nodes defined as $\mathcal{V} = \mathcal{V}^c \cup \mathcal{V}^l$. The set of camera and laser nodes are denoted with c and l respectively.



a. b. c.

Figure 7.2: Sensor graph constructed over a number of steps, introducing additional nodes (sensors) and edges at each. The simplest graph (a) has no shared relationship between node-pairs and therefore no loops — $\{\mathbf{g}_{c_1l_1}, \mathbf{P}_{c_1l_1}\} \in \mathcal{E}^{cl}$. In (b), a known measurement between the cameras is introduced as an edge where $\{\mathbf{g}_{c_1c_2}, \mathbf{P}_{c_1c_2}\} \in \mathcal{E}^{cc}$, and (c) represents the general case by extending the graph. The two types of edge used, $\mathcal{E}^{cl}$ and $\mathcal{E}^{cc}$, are indicated by red and blue arrows respectively.

For the purposes of this thesis we define two types of edge. The first connects a camera to a laser node and belongs to the set $\mathcal{E}^{cl}$, and whose value is determined by the camera–laser registration described in Chapter 5. The second connects two camera nodes, belongs to $\mathcal{E}^{cc}$, and constrains the upper-level optimisation.

The simplest graph contains only CL edges and has no redundancy. The case for two camera–laser pairs is shown in Figure 7.2a (two cameras to one laser).

Constraints

We define two types of constraints; the first is between the relative pose of adjacent camera–laser nodes $\mathbf{g}_{c_il_k}$ and its estimate $\hat{\mathbf{g}}_{c_il_k}$ with uncertainty $\mathbf{P}_{ik}$. Our formulation is typical for pose optimisation [29, 39]. Consider the sensor configuration in Figure 7.2a, which depicts independent calibrations of two cameras with a single laser scanner and for which $\mathbf{g}_{c_il_k} \in \mathbf{g}_{cl} \subseteq \mathcal{E}^{cl}$. We impose a relative pose constraint whose error is calculated as,

$$\mathbf{e}(\mathbf{g}_{c_il_k}; \hat{\mathbf{g}}_{c_il_k}) = \mathbf{g}_{c_il_k} \ominus \hat{\mathbf{g}}_{c_il_k}. \tag{7.1}$$

The second type of constraint is formulated by introducing a known measurement

between the two cameras, $\mathbf{g}_{c_i c_j}$. The constraint is formed using two camera nodes and a mutually adjacent laser node. The graph in Figure 7.2b depicts this configuration. To incorporate the constraint in the optimisation problem we express it as an error with respect to the known measurement $\bar{\mathbf{g}}_{c_i c_j}$, with edges (c_i, l_k) and (c_j, l_k) as

$$\begin{aligned} \mathbf{e}(\mathbf{g}_{c_i c_j}; \bar{\mathbf{g}}_{c_i c_j}) = \mathbf{e}(\mathbf{g}_{c_i l_k}, \mathbf{g}_{c_j l_k}; \bar{\mathbf{g}}_{c_i c_j}) = (\mathbf{g}_{c_i l_k} \ominus \mathbf{g}_{c_j l_k}) \ominus \bar{\mathbf{g}}_{c_i c_j}. \\ s.t. \quad \mathbf{g}^k_{c_i c_j} = \mathbf{g}_{c_i l_k} \ominus \mathbf{g}_{c_j l_k} \end{aligned} \tag{7.2}$$

The constraint in Equation (7.2) thus defines the edge of the graph such that $\mathbf{g}_{c_i c_j} \in \mathbf{g}_{cc} \subseteq \mathcal{E}^{cc}$.

The resulting error of each constraint is modelled as a Gaussian random variable and parameterised as $\mathbf{e}(\mathbf{g}_{c_i l_k}) \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_{c_i l_k})$ and $\mathbf{e}(\mathbf{g}_{c_i c_j}) \sim \mathcal{N}(\mathbf{0}, \bar{\mathbf{P}}_{c_i c_j})$, respectively. The squared error term used in the subsequent objectives is the Mahalanobis distance, given as $\|\mathbf{e}\|^2_{\mathbf{P}} = \mathbf{e}^\top \mathbf{P}^{-1} \mathbf{e}$.

7.1.3 Global objective function formulation

We begin by defining a global multi-objective function that jointly solves for all pose parameters in the specific case of the three-edge graph in Figure 7.2b. From this, we will then formulate the general case for any number of sensors. Recall that the objective function $D(\mathbf{g})$ from Section 5.1 measures the registration between camera and laser scanner. We will modify the notation such that the objective $D_{ik}(\mathbf{g}_{c_i l_k})$ registers camera i with laser k. The global optimisation is then given as follows,

$$\min_{\mathbf{g}_{c_1 l_1}, \mathbf{g}_{c_2 l_1}} \quad \lambda D_{1,1}\big(\mathbf{g}_{c_1 l_1}\big) + \lambda D_{2,1}\big(\mathbf{g}_{c_2 l_1}\big) + \|\mathbf{e}(\mathbf{g}_{c_1 l_1}, \mathbf{g}_{c_2 l_1}; \bar{\mathbf{g}}_{c_1 c_2})\|^2_{\bar{\mathbf{P}}_{c_1 c_2}}. \tag{7.3}$$

The pairwise objectives in Equation (7.3) have a non-zero scaling factor λ to adequately scale them against the squared error term. Solving this optimisation directly over all sensors is intractable, thus the problem is decoupled whereby each

sensor-pair calibration $\mathbf{g}_{c_i l_j}$ is solved for individually. The assumption that $D\big(\mathbf{g}_{c_1 l_1}\big)$ and $D\big(\mathbf{g}_{c_2 l_1}\big)$ are mutually independent allows for efficient and fast optimisation by decoupling the problem into sub-problems that can be solved in parallel.

To partition the parameter space, we introduce additional optimisation variables so that Equation (7.3) can be re-written as a function of these variables:

$$\begin{aligned} \min_{\substack{\mathbf{g}_{c_1 l_1},\, \mathbf{g}_{c_2 l_1},\\ \hat{\mathbf{g}}_{c_1 l_1},\, \hat{\mathbf{g}}_{c_2 l_1}}} \quad & \lambda D_{1,1}\big(\hat{\mathbf{g}}_{c_1 l_1}\big) + \lambda D_{2,1}\big(\hat{\mathbf{g}}_{c_2 l_1}\big) + \|\mathbf{e}(\mathbf{g}_{c_1 l_1}, \mathbf{g}_{c_2 l_1}; \bar{\mathbf{g}}_{c_1 c_2})\|^2_{\bar{\mathbf{P}}_{c_1 c_2}} \,. \\ \text{s.t.} \quad & \hat{\mathbf{g}}_{c_1 l_1} \stackrel{\text{def}}{=} \mathbf{g}_{c_1 l_1},\ \hat{\mathbf{g}}_{c_2 l_1} \stackrel{\text{def}}{=} \mathbf{g}_{c_2 l_1}. \end{aligned} \tag{7.4}$$

By 'terminating' the equality constraints in Equation (7.4) using quadratic penalty terms, we can derive the following unconstrained optimisation problem:

$$\begin{aligned} \min_{\substack{\mathbf{g}_{c_1 l_1},\, \mathbf{g}_{c_2 l_1},\\ \hat{\mathbf{g}}_{c_1 l_1},\, \hat{\mathbf{g}}_{c_2 l_1}}} \quad & \big\|\mathbf{e}\big(\mathbf{g}_{c_1 l_1}, \mathbf{g}_{c_2 l_1}; \bar{\mathbf{g}}_{c_1 c_2}\big)\big\|^2_{\bar{\mathbf{P}}_{c_1 c_2}} \\ & + \lambda D_{1,1}\big(\hat{\mathbf{g}}_{c_1 l_1}\big) + \|\hat{\mathbf{g}}_{c_1 l_1} \ominus \mathbf{g}_{c_1 l_1}\|^2_{\mathbf{P}_{c_1 l_1}} \\ & + \lambda D_{2,1}\big(\hat{\mathbf{g}}_{c_2 l_1}\big) + \|\hat{\mathbf{g}}_{c_2 l_1} \ominus \mathbf{g}_{c_2 l_1}\|^2_{\mathbf{P}_{c_2 l_1}} \end{aligned} \tag{7.5}$$

7.1.4 Decoupling the objective function

The following sub-sections describe how the calibration problem is partitioned, and how the objective functions for each level are defined and linked between iterations. We use the approach of [12] to alternate between the optimisation of sub-problems.

Alternating optimisation has shown to be useful for optimising problems in high-dimensional space [11], particularly if sub-problems allow for natural partitioning of variables. Such partitioning is natural for each pairwise calibration problem, yet the constraint in the first line of Equation (7.5) requires that the parameters be optimised jointly. The additional parameters as defined in Equation (7.4) allow for partitioning the variable space such that problem in Equation (7.5) can be effectively separated

into sub-problems such that each row is treated independently.

The optimisation is iterative, solving first the set of independent sub-problems of the lower-level. The upper-level is then solved by minimising the squared error over all pose variables. The system repeats until convergence.

Lower-level optimisation

The lower-level seeks to solve the pairwise registration for each sensor-pair independently. For every camera–laser pair $(i, k) : \{\mathbf{g}_{c_i l_k}, \mathbf{P}_{c_i l_k}\} \in \mathcal{E}^{cl}$, the optimisation for iteration n is given generally as

$$\min_{\hat{\mathbf{g}}_{c_i l_k}} \; \lambda D_{ik}\big(\hat{\mathbf{g}}_{c_i l_k}\big) + \big\|\mathbf{e}\big(\hat{\mathbf{g}}_{c_i l_k}; \mathbf{g}_{c_i l_k}\big)\big\|^2_{\mathbf{P}^n_{c_i l_k}} . \tag{7.6}$$

Note that Equation (7.6) is a sub-problem decoupled from the second or third row of Equation (7.5), depending on the camera index. While solving this problem, the variable tied to the upper-level, the hatless $\mathbf{g}_{c_i l_k}$, is held constant.

The lower-level of Equation (7.6) is partitioned in a way that all sub-problems are treated independently. Each sub-problem thus seeks only to solve the registration of a particular camera and laser pair, the formulation of which is covered in detail in Chapter 5. This partitioning also allows for efficient parallel processing of each registration problem. The set of solutions $\hat{\mathbf{g}}_{cl}$ produced by the lower-level is used to drive the constraints of the upper-level pose optimisation.

Upper-level optimisation

For the problem defined in Equation (7.5), the lower-level optimiser provides solutions of $\hat{\mathbf{g}}_{c_1 l_1}$ and $\hat{\mathbf{g}}_{c_2 l_1}$. Conversely, the upper-level holds the hatted variables, $\hat{\mathbf{g}}_{c_1 l_1}$ and $\hat{\mathbf{g}}_{c_2 l_1}$, as known constants. The optimisation is a combination of the error terms from Equation (7.5) as functions of the un-hatted variables $\mathbf{g}_{c_i l_j}$, and is formulated as

$$\begin{aligned}\min_{\substack{\mathbf{g}_{c_1 l_1},\\ \mathbf{g}_{c_2 l_1}}} \quad & \left\|\mathbf{e}\big(\mathbf{g}_{c_1 l_1}, \mathbf{g}_{c_2 l_1}; \bar{\mathbf{g}}_{c_1 c_2}\big)\right\|^2_{\bar{\mathbf{P}}_{c_1 c_2}} \\ & + \left\|\mathbf{e}\big(\mathbf{g}_{c_1 l_1}; \hat{\mathbf{g}}_{c_1 l_1}\big)\right\|^2_{\mathbf{P}^n_{c_1 l_1}} \\ & + \left\|\mathbf{e}\big(\mathbf{g}_{c_2 l_1}; \hat{\mathbf{g}}_{c_2 l_1}\big)\right\|^2_{\mathbf{P}^n_{c_2 l_1}}\end{aligned} \tag{7.7}$$

Herein, the solutions of the lower-level optimisers drive the upper-level optimiser, and vice versa. The problem in Equation (7.7) is specific to the configuration depicted in Figure 7.2b, where two cameras are calibrated to a single laser scanner. To generalise this problem, we identify and sum the error of all available camera–camera measurements (top row), and independent camera–laser error terms (second and third rows). Summing the errors of both constraints from Equations (7.1) and (7.2) we obtain

$$\min_{\mathbf{g}_{cl}} \quad \sum_{i,j,k} \left\|\mathbf{e}\big(\mathbf{g}_{c_i l_k}, \mathbf{g}_{c_j l_k}; \bar{\mathbf{g}}_{c_i c_j}\big)\right\|^2_{\bar{\mathbf{P}}_{ij}} + \sum_{i,k} \left\|\mathbf{e}\big(\mathbf{g}_{c_i l_k}; \hat{\mathbf{g}}_{c_i l_k}\big)\right\|^2_{\mathbf{P}^n_{ik}}. \tag{7.8}$$

The lower- and upper-level problems are thus captured by Equations (7.6) and (7.8), respectively. The upper-level problem is formulated as a pose optimisation and as such, forms a standard non-linear least squares problem. We can therefore solve this using the Levenburg-Marquardt algorithm as covered in Section 4.6. The Ceres Solver package [1], a feature-rich non-linear solver written in C++, offers an efficient implementation. The set of solutions $\mathbf{g}_{cl}$ is then used to initiate the next iteration, beginning again with the lower-level.

Covariance

The covariance $\mathbf{P}^n_{cl}$ (and $\mathbf{P}^n_{vl}$) represents the uncertainty in a laser–camera calibration estimate at the n^{th} iteration of the optimisation. On suggestion [10], we choose a decaying value such that the covariance is a function of the iteration, given as

$$\mathbf{P}^n_{cl} = \omega^n \mathbf{P}^0_{cl}. \tag{7.9}$$

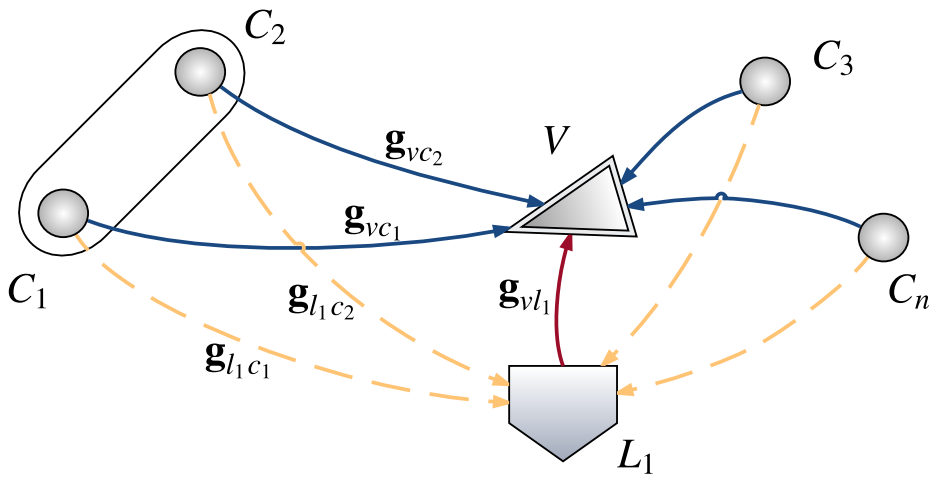


Figure 7.3: Graph of the sensor network, accounting for ego-motion. The transforms indicated by the solid blue lines indicate the fixed, known constants $\bar{\mathbf{g}}_{vc}$. The dashed lines indicate derived values $\mathbf{g}_{l_kc_i}$ not explicitly measured or solved. For example, $\mathbf{g}_{l_1c_1}$ is derived from $\bar{\mathbf{g}}_{vc_1}$ and the estimate $\hat{\mathbf{g}}^1_{vl_1}$ from the lower-level.

For experimentation, we choose a sufficiently large covariance, $\sigma_t^2 = 1\text{m}$ and $\sigma_\phi^2 = 1\text{rad}$ giving the initial covariance as the diagonal matrix

$$\mathbf{P}^0 = \begin{bmatrix} \boldsymbol{I}_3\sigma_t^2 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I}_3\sigma_\phi^2 \end{bmatrix}.$$

If the covariance is too small, the error terms of Equation (7.6) will dominate. If too large, the upper-level optimisation favours the $\hat{\mathbf{g}}$ measurements. The scalar, λ, works to balance this trade-off.

Accounting for ego-motion

The formulation of Equations (7.6) and (7.8) above is suitable if the point cloud is stationary, or if the camera to which the laser is being calibrated has the same reference frame as the vehicle. The latter case presumes that the camera is thus also the source of odometry. If a pose source other than visual odometry is used (such as an INS), the vehicle frame needs to be made explicit. We use *vehicle* frame to mean

centre of ego-motion.

The system $\{C_1, C_2, L_1, V\}$ in Figure 7.3 represents the same system as that in Figure 7.2b, with the added intermediary V. Recalling Equation (4.2),

$$ {}^{c^a}\mathbf{p} = \mathbf{g}_{cv} \oplus \mathbf{g}_{v^a v^b} \oplus \mathbf{g}_{vl} \oplus {}^{l^b}\mathbf{p}, $$

where the point ${}^{c^a}\mathbf{p}$ is a function of two potentially unknown sensor poses, $\mathbf{g}_{vc}$ and $\mathbf{g}_{vl}$. Our experimentation involves the use of a binocular stereo camera, which provides the identity between the first camera and the vehicle frame, and an accurately known pose of the second camera. This is the ideal case, but the formulation allows for any camera of a known pose.

The constraint in Equation (7.1) imposed by the lower-level presumes the above case where the pose $\hat{\mathbf{g}}_{cl}$ is solved for, independent of vehicle frame. Instead, we abstract the vehicle frame such that the lower-level solves for $\hat{\mathbf{g}}_{vl}$, and the measurements $\bar{\mathbf{g}}_{cc}$ in Equation (7.2) become $\bar{\mathbf{g}}_{vc}$ to constrain the upper-level. The set of lower-level objectives can thus be rewritten as

$$ \min_{\hat{\mathbf{g}}^i_{vl_k}} \; \lambda D_{ik}\big(\hat{\mathbf{g}}^i_{vl_k}, \bar{\mathbf{g}}_{vc_i}\big) + \Big\|\mathbf{e}(\mathbf{g}_{vl_k}; \hat{\mathbf{g}}^i_{vl_k})\Big\|^2_{\mathbf{P}^n_{vl_k}} . \tag{7.10} $$

The corresponding constraint is formulated accordingly,

$$ \mathbf{e}(\mathbf{g}_{vl_k}; \hat{\mathbf{g}}^i_{vl_k}) = \mathbf{g}_{vl_k} \ominus \hat{\mathbf{g}}^i_{vl_k}. \tag{7.11} $$

The solution of Equation (7.10) will be a set of solutions for $\hat{\mathbf{g}}^i_{vl_k}$, one for each camera–laser pair $\{i, k\}$. If we simply find the maximum-likelihood estimate of these solutions (minimising the sum of the errors in Equation (7.11) over all pairs $\{i, k\}$), we no longer have a pose constraint imposed by the cameras. Notice from Figure 7.3 that we can represent $\mathbf{g}^i_{vl_k}$ with respect to the camera frame since $\bar{\mathbf{g}}_{vc_i}$ is known constant. We thus constrain the camera–laser using

$$
\begin{aligned}
\mathbf{e}\big(\mathbf{g}_{c_i l_k}; \hat{\mathbf{g}}_{c_i l_k}\big) = \mathbf{e}\big(\mathbf{g}_{vc_i}, \mathbf{g}_{vl_k}; \bar{\mathbf{g}}_{vc_i}, \hat{\mathbf{g}}^i_{vl_k}\big) \quad &= (\ominus \mathbf{g}_{vc_i} \oplus \mathbf{g}_{vl_k}) \ominus (\ominus \bar{\mathbf{g}}_{vc_i} \oplus \hat{\mathbf{g}}^i_{vl_k}) \\
s.t. \quad \mathbf{g}_{c_i l_k} &= (\ominus \mathbf{g}_{vc_i} \oplus \mathbf{g}_{vl_k}) \\
\hat{\mathbf{g}}_{c_i l_k} &= (\ominus \bar{\mathbf{g}}_{vc_i} \oplus \hat{\mathbf{g}}^i_{vl_k})
\end{aligned}
\tag{7.12}
$$

The upper-level objective can now be reformulated by combining the sums of the errors in Equation (7.12), and the known measurements $\bar{\mathbf{g}}_{vc}$ to obtain

$$
\min_{\mathbf{g}_{vl},\ \mathbf{g}_{vc}} \quad \sum_i \left\| \mathbf{e}\big(\mathbf{g}_{vc_i}; \bar{\mathbf{g}}_{vc_i}\big) \right\|^2_{\bar{\mathbf{P}}_{vc_i}} + \sum_{i,k} \left\| \mathbf{e}\big(\mathbf{g}_{vl_k}, \mathbf{g}_{vc_i}; \hat{\mathbf{g}}_{c_i l_k}\big) \right\|^2_{\mathbf{P}^n_{ik}} \tag{7.13}
$$

The covariance $\mathbf{P}^n_{ik}$ is the combined uncertainty of $\mathbf{P}^n_{vl_k}$ and $\bar{\mathbf{P}}_{vc_i}$. We follow [89] to find the resulting Gaussian uncertainty $\mathbf{P}_{ac}$ from the operation $\mathbf{g}_{ac} = \mathbf{g}_{ab} \oplus \mathbf{g}_{bc}$. The approximation, which is accurate to first-order, is given as

$$
\begin{aligned}
\mathbf{P}_{ac} &= \mathbf{P}_{bc} + Ad(\ominus \mathbf{g}_{ab}) \mathbf{P}_{ab} Ad(\ominus \mathbf{g}_{ab})^\top \\
\text{where} \quad Ad(\mathbf{g}) &= \begin{bmatrix} \mathbf{R}(\boldsymbol{\phi}) & \lfloor \mathbf{t} \rfloor_\times \mathbf{R}(\boldsymbol{\phi}) \\ \mathbf{0} & \mathbf{R}(\boldsymbol{\phi}) \end{bmatrix}.
\end{aligned}
\tag{7.14}
$$

The problems in Equations (7.10) and (7.13) present the lower- and upper-level of the alternating calibration problem reformulated to account for ego-motion through an intermediary reference frame V.

7.2 Results

The system we tested on comprised of a laser scanner, stereo camera (two cameras) and two mono cameras. The testing setup used is highlighted in Figure 7.6. We present the results in Figures 7.4 and 7.5, which show the calibration error

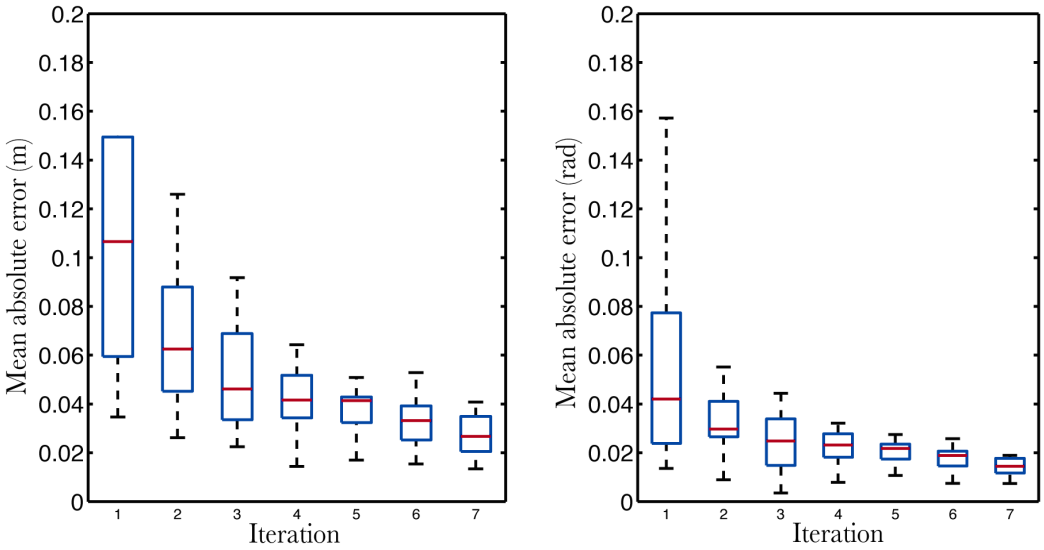


Figure 7.4: Error plot of the alternating calibration on Milton Keynes dataset, of Bumblebee stereo camera (left) to laser. The left-hand graph shows the mean absolute error of the norm of translation, while the right-hand graph shows the mean error in rotation, both per iteration of the optimisation. Each lower-level objective combines 20 scenes.

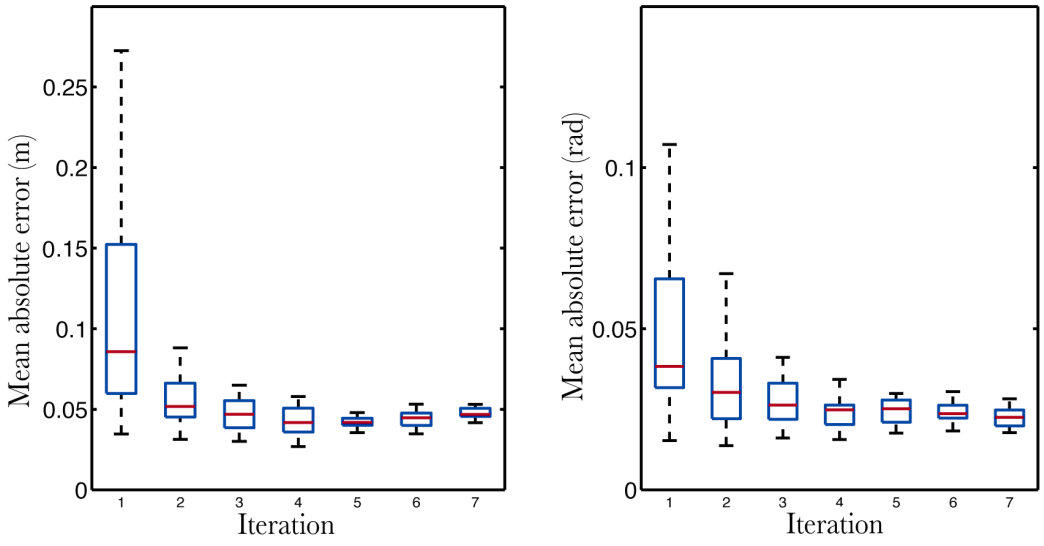


Figure 7.5: Error plot of the alternating calibration on Milton Keynes dataset, of Grasshopper mono camera to laser. The left-hand graph shows the mean absolute error of the norm of translation, while the right-hand graph shows the mean error in rotation, both per iteration of the optimisation. Each lower-level objective combines 20 scenes.

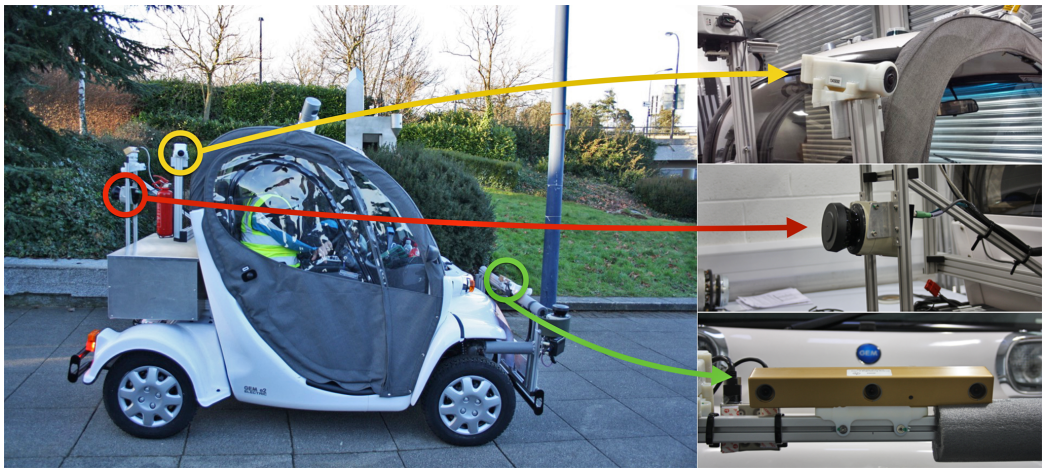


Figure 7.6: The sensor configuration used for experimentation. We use the stereo camera mounted at the front of the vehicle to estimate vehicle pose using VO. Two wide-angle mono cameras (one not shown) capture the surroundings on either side of the vehicle, and a single 2D laser scanner is used to generate point cloud swathes.

($\|\hat{\mathbf{g}}_t \ominus \mathbf{g}_t^{truth}\|$ and $\|\hat{\mathbf{g}}_\phi \ominus \mathbf{g}_\phi^{truth}\|$) for the left stereo camera to the laser scanner, and mono camera to laser scanner, respectively.

We measure the performance of calibration using a Monte-Carlo statistical approach. The alternating calibration was run N times, each from a different initial seed, to measure the calibration and standard deviation at each iteration. The error is calculated with respect to the ground truth values obtained using the method described in Section 4.7.3. Unlike the reporting of error in other sections, we cannot use the stereo baseline for comparison as it is used as a constraint in the optimisation; baseline error will tend to zero by design.

Figures 7.4 and 7.5 present the results for the left stereo-camera sensor and mono camera respectively. The box plots represent the distribution of error at each iterations. In each case, the magnitude of the error, with respect to *ground truth*, decreases with each iteration. The first iteration, when the pairwise NID cost is most dominant, represents the calibration solution if the alternating method is not followed as this is the solution as obtained in Section 5.1.3.

We also see that the values of $\mathbf{g}$ and $\hat{\mathbf{g}}$ converge; Figure 7.8 shows the evolution of the lower-level and upper-level solutions. The mechanism that achieves this is made

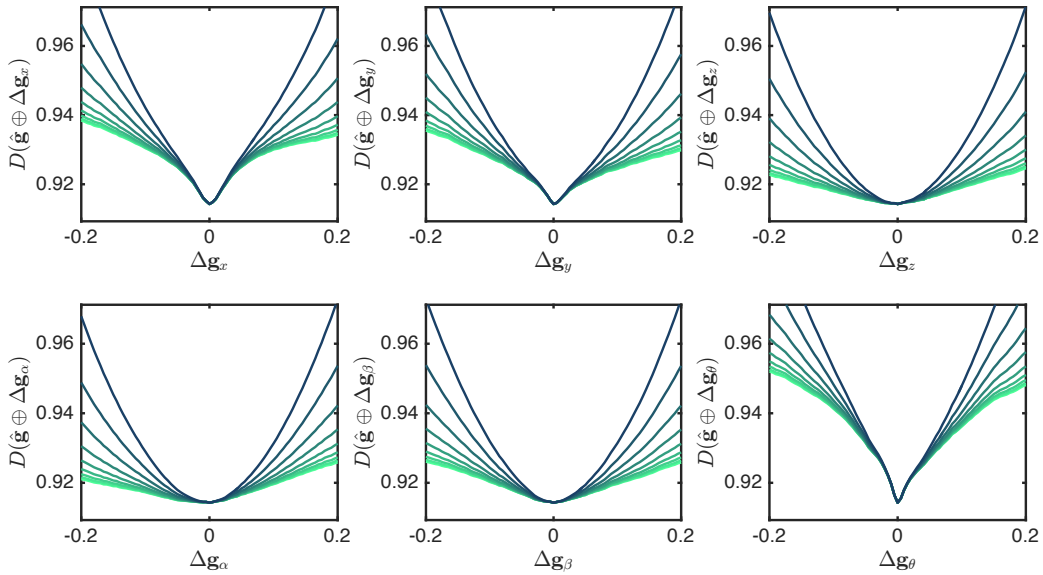


Figure 7.7: Cost surface evolution for the lower-level objective function. The cost surface is plotted for each 6-DoF parameter of the objective function in Equation (7.6), for the first 8 iterations (green $\rightarrow$ blue). The parabolic shape is due to the increasing dominance of the error term, $\mathbf{e}^\top \mathbf{P}^{-1} \mathbf{e}$, as the covariance decreases according to Equation (7.9).

clear in Figure 7.7, which shows the cost surface of the lower-level at iterations 1 to 8. The parabolic shape of the cost surface is predicated on the lower covariance calculated by Equation (7.9).

Finally, we show qualitative evidence that solutions obtained successfully register the camera to the laser point cloud. The matrix of images in Figure 7.9 shows the camera edge image overlain on a synthetic laser image. The initial calibration shows an obviously incorrect camera–laser registration. Our alternating method of calibration shows registration comparable to that obtained from 'ground truth' measurements.

7.3 Conclusions

In preceding chapters we formulated the pairwise calibration of a camera and laser scanner as an independent system. Treated as such, there is no provision should the calibration of this pairwise system fail, be ill-constrained or otherwise inaccurate. The calibration method based on the Normalised Information Distance in Section 5.1

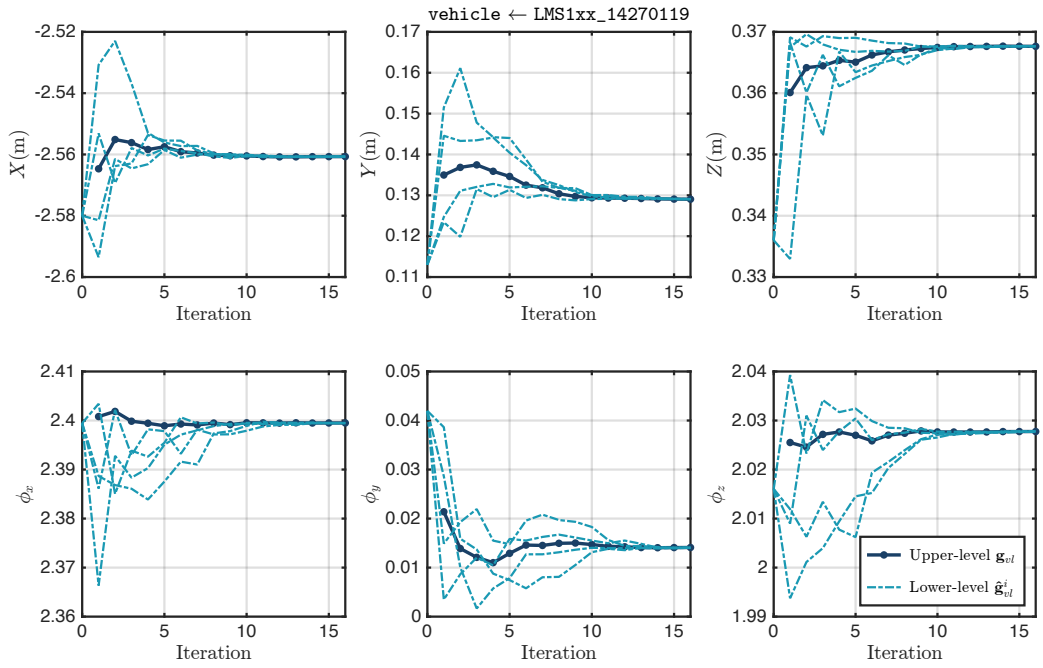


Figure 7.8: Convergence of lower- and upper-level solutions for four cameras on the GEM, plotted for each of the 6-DoF parameters independently. Translation (top row) is in meters while rotation is in axis-angle representation. The solid line indicates the evolution of the upper-level solution $\mathbf{g}_{vl}$, while the dashed lines indicate the lower-level solution of $\hat{\mathbf{g}}^i_{vl}$ for each of the four cameras.

demonstrated inconsistent convergence from different initial seeds. This suggests that there are multiple local minima even for a given set of scenes.

By realising that a robot has a network of sensors, we proposed a unified approach to calibration that mitigates the tendencies for divergent optimisation and improves overall calibration accuracy. The behaviour of 'correcting' errant lower-level calibration is evident in Figure 7.8.

Since we presume that the computational cost of the pairwise registration is large, optimising jointly over the entire parameter space would be intractable. Instead we reformulate the objective function as to decouple the expensive registration from the graph constraints. Introducing known measurements allowed us to formulate pose constraints to close the network, thus presenting an optimisation over all sensor parameters simultaneously. To address the issue of a high–dimensional parameter space, we use alternating optimisation to treat each camera–laser registration independently,

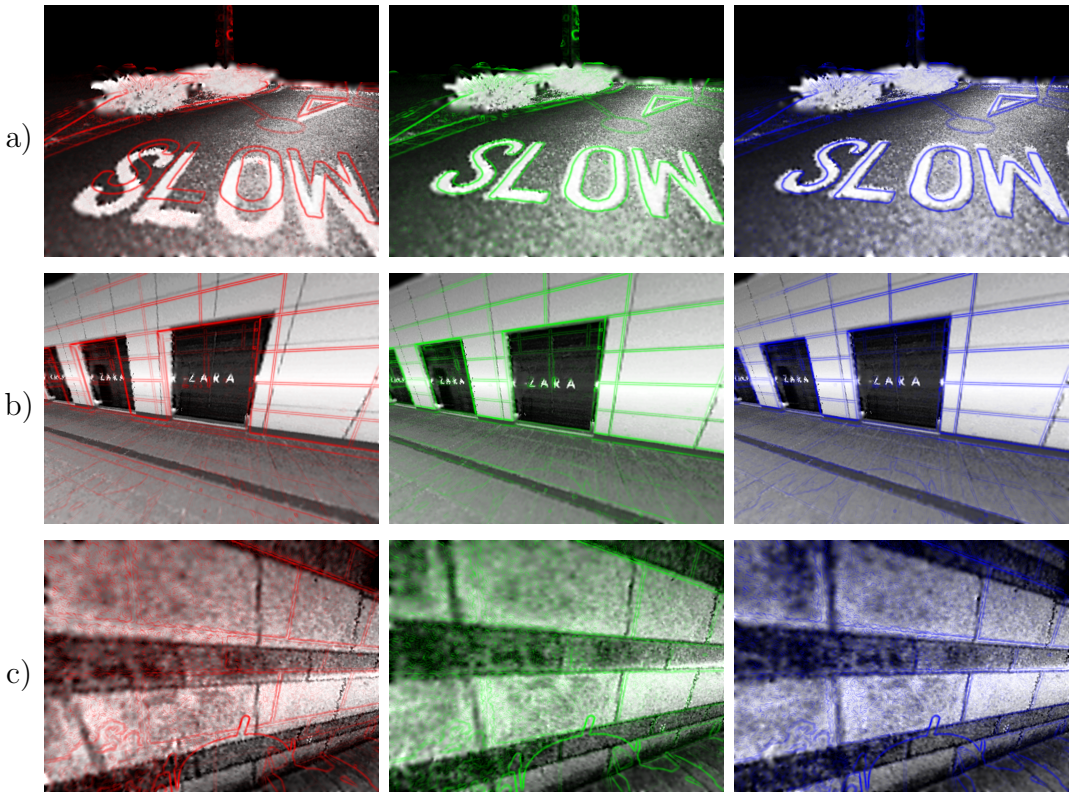


Figure 7.9: Laser–camera overlay, comparison of visual alignment. Edges of a scene image superimposed on a synthetic image constructed using a projected point cloud. Rows correspond to a different camera used; a) front stereo camera, b) right mono camera, and c) left mono camera. The columns correspond to the calibration parameters of: a random initial seed, using our proposed framework and of the 'ground truth', respectively.

in what we term the *lower-level*, solving only those respective pose parameters. The solutions from the lower-level then drive the pose constraints of *upper-level* pose graph.

We demonstrate the alternating method on the GEM platform, having a sensor configuration consisting of two mono-cameras, a stereo camera, and a 2D laser scanner. Overall, calibration converges consistently to a value closer to ground-truth than a single pairwise registration (Section 5.1.3) can achieve. There are however areas of improvement, particularly as we only consider the case of cameras and lasers. We conclude this chapter by discussing areas of potential future research.

7.3.1 Future work

One of the core assumptions we make is about the availability of accurate camera–camera calibration (or other such constraining measurements). A stereo camera is often a core component of the robotic systems, and for which this assumption is safe. It is also an added convenience that the stereo-camera is the centre of ego-motion. The mono-cameras, however, are calibrated using the motion tracker as described in Section 4.7.3 and therefore do not offer the same consistency or accuracy of calibration as one might expect from a self-contained stereo unit.

We have mitigated the effect of an incorrect calibration of the mono-cameras by specifying a much higher covariance than that of the stereo unit, yet this error still propagates through the pose graph, affecting the whole system.

As such, there are two particular scenarios that the current implementation of alternating calibration cannot account for:

1. The calibration of cameras relative to the pose source is unknown, or known with large uncertainty.

2. A stereo camera is not used as a pose source.

The latter is essentially a particular case of the former, whereby the vehicle–camera calibration is not known for any camera. The issue is compounded when one realises that the metric accuracy and consistency of the laser point cloud is dependent on the vehicle–laser calibration. For a given camera and laser pair, there are two unknown transforms $\mathbf{g}_{vc}$ and $\mathbf{g}_{vl}$. This is clear from Figure 7.3.

The mutual dependence of $\mathbf{g}_{vc}$ and $\mathbf{g}_{vl}$ could potentially be exploited in a particular way if there are multiple cameras in the system. Errors in the point cloud (due to error in $\mathbf{g}_{vl}$) could be more apparent when projected to a different camera, thus providing a more 'corrective' calibration.

It is not obvious what effect the parameter partitioning we have chosen has on the overall optimisation. Further experimentation of the alternating method would

involve choosing partitions that might best avoid the effect of slack parameters introduced by the intermediary *vehicle* frame. For example, does alternating between rotation and translation parameters have a meaningful effect?

This thesis has considered the calibration particularly of cameras and lasers, yet our unified calibration can naturally be extended to incorporate camera–camera and camera–INS pose constraints as a lower-level objective. We envision a framework that accounts for the eventualities discussed above, and that exploits known accurate calibrations where available.

Chapter 8

Conclusion and future work

This chapter provides a summary of the contributions of this thesis and recommendations for future work. The contributions fall into two distinct categories; the design of a reconfigurable portable mapping device (Chapter 2), and the automatic calibration of cameras and 2D laser scanners, particularly of those that do not necessarily have overlapping fields of view (Chapters 5 to 7).

8.1 Summary of contributions

This thesis opened with the **design, construction and testing of a portable and cost efficient surveying device** for research or industry purposes. The *NABU 3*, the third generation mapping device developed at the ORI, is capable of producing high quality 3D scene reconstructions. Taking advantage of the flexibility of 3D printing, the *NABU 3* was designed to be modular and reconfigurable, offering the user the ability to replace sensors and modify sensor positions for optimal mapping, or for demonstrating particular robotics autonomy systems. As a testament to its capability the *NABU 3* and subsequent adaptations are in continual use by the ORI and Oxbotica Ltd.

The modularity of the device lends itself naturally to challenges of obtaining and

maintaining accurate calibration for all sensors. The remainder of the contributions pertained to the goal of creating a fully automatic sensor-suite calibration framework, not limited to the use on the *NABU 3*.

We provided an **analysis of normalised information distance for calibration** in Section 5.1. Information theoretic measures for camera–laser calibration using natural scenes have been used in the past [62, 81, 88] with a high degree of success. We extended their work to the domain of 2D laser scanners, which do not necessarily offer overlapping fields of view with respective cameras. By accumulating subsequent scans through motion of the vehicle, we can simulate this necessary overlap.

We conclude that, in isolation, the NID method does not provide a robust metric for camera–laser registration since anomalous scene geometry and appearance can adversely affect the registration. Pre-processing raw laser and camera appearance data (histogram equalisation and Gaussian smoothing) has shown to significantly improve registration performance (Section 5.1.2). Ultimately, we found that registration is predicated on observing scenes that offer geometry and appearance conducive to using the NID metric. The advantages of using an information measure for registration are however significant and desirable:

- It offers a metric that is agnostic about the spatial relation between individual laser points and is thus not affected by their relative sparsity.
- Explicit data association is not required, ideally alleviating the need for manual intervention.
- As a corollary to the previous point, calibration targets are not required. We use the robot's observations during its normal intended operation.

The versatility of the NID metric inspired **a policy of scene selection to improve accuracy and reliability of calibration** (Chapter 6). We considered three approaches that select likely suitable scenes:

1. Classifying scenes according to a learnt model of the shape of the cost surface around the solution from successful calibrations [76], Section 6.2.2.

2. Characterising the shape of the cost surface using measures of *smoothness* and *distinctness*, Section 6.2.3.

3. Classifying scenes according to a learnt model of image features from successful calibrations, Section 6.4.1.

We identified in (1) that posing the scene selection problem as a binary classification is unsuitable. In addition, acquiring meaningfully labelled training data presents its own challenge. We instead proposed a *scene score* (2) which attempts to measure the efficacy of a scene based on its resulting cost surface. By choosing scenes that score highly according to this measure, we were able to eliminate failed registrations and greatly improve on accuracy and precision using only a few scenes.

The scene score requires sampling from the cost function around the minimum. Depending on the number of samples this can be an expensive operation. A feature-based scoring mechanism that uses only the observations as inputs would be favourable. Our preliminary findings in (3) show that image-based features alone provide little discriminatory power using rudimentary implementations. We provide more detail on possible future work in Section 8.2.

In addition to the NID formulation, in Section 5.2 we proposed a **calibration algorithm that uses automatically extracted cross-modal point features**. We demonstrated that SIFT features [46] can be successfully matched across appearance modalities using even a low-fidelity laser scanner such as the Hokuyo UTM-30LX, albeit with a very low true-positive recall. The abundance of real-world data allowed us to be conservative in the cross-matching of features. These matches were then used to derive a hard data association between a pixel location and 3D laser point. The registration was solved by minimising the projection error over all matched points.

The results showed successful camera–laser calibration across all testing platforms. The performance is also shown to be superior to the NID method for the warehouse dataset. This particular dataset is small in comparison to the others, providing less of an opportunity to find good scenes according to our *scene selection* policy. Significantly, the point feature method is innately robust to appearance anomalies such as overexposure or shadows, ultimately eschewing the need for intelligent scene selection.

Finally, we proposed a **unifying calibration framework for a system of sensors** in Chapter 7. We formulated a multi-objective function incorporating multiple sensor–pair registrations and using prior known camera–camera calibration to constrain a pose graph. Optimising jointly over the entire parameter space would be intractable as we presume the pairwise registration to be an expensive black box operation.

By partitioning the parameter space, the multiple objectives can be solved in an alternating manner. The partition is performed such that we can separate the problem into two levels. The lower-level independently solves each pairwise sensor registration while the upper-level solves the pose graph independent of the pairwise registration.

We demonstrated decreased system-wide calibration error on the GEM, which comprised of a stereo camera unit, two mono cameras and planar laser scanner. The upper-level pose graph acts to correct errant lower-level solutions, thus working to improve calibration consistency. We discuss possible future improvements in the following section.

8.2 Future work

This thesis is concluded by summarising the possible future work.

8.2.1 Portable Mapping

The *NABU 3* ultimately operated as intended, with few features that we would recommend significant modification. Since the device is a prototype, it would benefit from a refinement of existing features.

The splined sensor interfaces are the more experimental aspects of the *NABU*'s design, offering reconfigurability and modularity. These features were designed specifically for manufacture using 3D printing technology, as traditional methods would require expensive machining. We found that the quality of the surface finish and material tolerances meant that the precise fit for the splined mating parts was difficult to predict. We would therefore recommend simplifying the design of the interface as be to less prone to manufacturing uncertainties.

While the reconfiguration is an advantage, managing and supplying the correct calibrations for a given configuration remains a challenge. A solution, albeit rather involved, would be to automatically detect in which configuration (spline location) the sensor is seated. Even if reseating the sensor causes miscalibration, the stored calibration will provide a sufficiently close estimate as to be refined by the calibration algorithms.

General areas of refinement could include, but are not limited to:

- Waterproofing the chassis for all-weather operation.
- Redesign of the shell as to support the weight of the device should a handle be attached to the top. Currently, only the underneath of the chassis has been designed to bear load.

- The sensor interfaces are difficult to reconfigure as hand-room in the the chassis is limited.
- The side sensor mounts protrude significantly, compromising rigidity and making them prone to breaking.

8.2.2 Calibration

In this thesis we propose a general framework for calibrating a suite of sensors, with particular focus on the calibration of cameras to lasers. While the research presented here makes significant headway to a fully automatic and encapsulated calibration system, it falls short. We make recommendations that we believe will further progress towards this goal.

Unified calibration using alternating optimisation

- **Account for cameras with unknown calibration.** We have assumed that the network of cameras are cross-calibrated with a known and constant value. We thus are left to solve for the calibration of the laser scanner with respect to the vehicle. A more realistic scenario may be that not all camera calibrations are known. It would be worthwhile to investigate how the alternating calibration could be adapted to solve for the camera–vehicle calibration, and how this affects the system as a whole.
- **Experiment with using a pose source other than the stereo-camera.** The sensor configurations on all our robotic platforms use a stereo-camera for pose estimation. Conveniently, the centre of ego-motion is thus also the centre of the camera image.
- **Incorporate camera–camera and camera–INS registration to the lower-level.** The lower-level objective function is formulated such that it is able to

accommodate any pairwise cost function. Incorporating camera–camera calibration using motion [32] (since fields of view are assumed no to overlap) would likely benefit the system.

- **Fuse NID and point-features.** Robotics navigation systems frequently fuse sensor data from multiple sources to improve localisation and object detection. The same should be applied to calibration to ensure consistent and accurate measurements, particularly if the methods can be used such they complement each other.

Camera–laser calibration using NID, and scene selection

- **Experiment using modalities other than reflectance and pixel luminance.** Depending on the nature of the scene, selecting a modality other than those we have experimented with may improve robustness to appearance anomalies [96].

- **Investigate more robust formulations of the joint distribution.** The joint distribution in our implementation is modelled as a histogram using pixel luminance determined using bi-linear interpolation. Techniques include B-Spline approximation of the joint distribution [23] and spatial weighting [20].

- **Combine point cloud and image features.** We have only considered the discriminatory power of image features in determining the 'efficacy' of a particular scene. Incorporating 3D features [67] could potentially prove fruitful for use in machine learning methods.

- **Convolutional neural nets for feature learning.** Typically requiring huge datasets, CNNs can be supplemented with a pre-trained network, such as that trained on ImageNet [21] for classification [38], to supply robust image descriptors [17].

- **Unsupervised learning of suitable scenes.** The most significant challenge of using machine learning approaches was to supply labelled training data. A binary classification label is unsuitable and there is no impartial metric to score scenes. The *scene scoring* method of Chapter 6 offers a heuristic based on the shape of the resulting cost function, yet this score is by no means necessarily representative.

- **Discover regions of the scene that provide strong features for calibration.** The cameras (especially the front-facing stereo camera) predominantly view the ground plane, which means that not all 6-DoF parameters are well constrained by the scene geometry. Discovering regions which have strong features, and also diversify the 3D geometry will be valuable for reliable calibration in all 6-DoF parameters. Equally, regions that are high in visual artefacts that affect calibration negatively should be excluded.

Camera–laser calibration using automatic point features

- **Optimise feature extraction for use on laser reflectance images.** The SIFT feature extractor is an unmodified implementation of [85]. While already naturally robust to slight appearance changes, modifying the parameters and feature representation will likely yield more robust matches [75].

- **Robust outlier detection.** We use a functional yet rudimentary outlier rejection process that relies on the assumption that positive feature matches will be proximal. For greater consistency and resistance to larger initial miscalibration, an improved outlier rejection model will be required.

In this thesis we endeavoured to create unified framework for calibration that can be applied across platforms. Our focus has been on the use of cameras and laser scanners, yet the framework could easily be extended to incorporate other sensors. The number of mobile robots in operation will continue to increase, emphasising the

need for end-to-end autonomy in all aspects of robotics. We envision a policy of online calibration which, during the vehicle's normal operation, can reliably monitor and correct errant calibration for safer and more efficient autonomy.

Bibliography

[1] Sameer Agarwal, Keir Mierle, and Others. Ceres solver. `http://ceres-solver.org`.

[2] Hatem Alismail, L Douglas Baker, and Brett Browning. Automatic calibration of a range sensor and camera system. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on*, pages 286–292. IEEE, 2012.

[3] J.M. Alvarez, T. Gevers, and A.M. Lopez. 3d scene priors for road detection. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 57–64, 2010.

[4] Ian Baldwin. *Large-Scale Urban Localisation with a Pushbroom LIDAR*. PhD thesis, University of Oxford, Oxford, United Kingdom, 2012.

[5] Ian Baldwin and Paul Newman. Road vehicle localization with 2d push-broom lidar and 3d priors. In *Robotics and automation (ICRA), 2012 IEEE international conference on*, pages 2611–2617. IEEE, 2012.

[6] Arindam Banerjee, Inderjit S Dhillon, Joydeep Ghosh, and Suvrit Sra. Clustering on the unit hypersphere using von mises-fisher distributions. *Journal of Machine Learning Research*, 6(Sep):1345–1382, 2005.

[7] Sid Yingze Bao, Min Sun, and Silvio Savarese. Toward coherent object detection and scene layout understanding. *Image and Vision Computing*, 29(9):569 – 579, 2011.

[8] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.

[9] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*, pages 404–417. Springer, 2006.

[10] D. P. Bertsekas. The method of multipliers for equality constrained problems. In *Constrained Optimization and Lagrange Multiplier Methods*, pages 95–157, 1996.

[11] James C Bezdek and Richard J Hathaway. Some notes on alternating optimization. In *AFSS International Conference on Fuzzy Systems*, pages 288–300. Springer, 2002.

[12] James C Bezdek and Richard J Hathaway. Convergence of alternating optimization. *Neural, Parallel & Scientific Computations*, 11(4):351–368, 2003.

[13] Jose-Luis Blanco. A tutorial on se (3) transformation parameterizations and on-manifold optimization. *University of Malaga, Tech. Rep*, 2010.

[14] Jan Böhm and Susanne Becker. Automatic marker-free registration of terrestrial laser scans using reflectance. In *Proceedings of 8th Conference on Optical 3D Measurement Techniques, Zurich, Switzerland*, pages 338–344, 2007.

[15] Michael Bosse, Robert Zlot, and Paul Flick. Zebedee: Design of a spring-mounted 3-d range sensor with application to mobile mapping. *IEEE Transactions on Robotics*, 28(5):1104–1119, 2012.

[16] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *European conference on computer vision*, pages 778–792. Springer, 2010.

[17] Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*, 2014.

[18] Winston Churchill and Paul Newman. Continually improving large scale long term visual navigation of a vehicle in dynamic urban environments. In *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pages 1371–1376. IEEE, 2012.

[19] Mark Cummins and Paul Newman. Appearance-only slam at large scale with fab-map 2.0. *The International Journal of Robotics Research*, 30(9):1100–1123, 2011.

[20] Amaury Dame and Eric Marchand. Mutual information-based visual servoing. *IEEE Transactions on Robotics*, 27(5):958–969, 2011.

[21] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[22] Julie Dequaire, Chi Hay Tong, Winston Churchill, and Ingmar Posner. Off the beaten track: Predicting localisation performance in visual teach and repeat. In *Robotics and Automation, IEEE International Conference on*, 2016.

[23] Nicholas Dowson and Richard Bowden. A unifying framework for mutual information methods for use in non-linear optimisation. In *European Conference on Computer Vision*, pages 365–378. Springer, 2006.

[24] Russ C Eberhart and James Kennedy. A new optimizer using particle swarm theory. In *Proceedings of the sixth international symposium on micro machine and human science*, volume 1, pages 39–43. New York, NY, 1995.

[25] Christian Forster, Luca Carlone, Frank Dellaert, and Davide Scaramuzza. On-Manifold Preintegration Theory for Fast and Accurate Visual-Inertial Navigation. *arXiv:1512.02363 [cs]*, December 2015. 00000.

[26] Chao Gao and John R Spletzer. On-line calibration of multiple lidars on a mobile vehicle platform. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 279–284. IEEE, 2010.

[27] Gwennael Gate, Amaury Breheret, and Fawzi Nashashibi. Centralized fusion for fast people detection in dense environment. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 76–81, 2009.

[28] Xiaojin Gong, Ying Lin, and Jilin Liu. 3d lidar-camera extrinsic calibration using an arbitrary trihedron. *Sensors*, 13(2):1902–1918, 2013.

[29] Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard. Nonlinear constraint network optimization for efficient map learning. *IEEE Transactions on Intelligent Transportation Systems*, 10(3):428–439, 2009.

[30] Alastair Harrison. *Precision Timing and Measurement for Inference with Laser and Vision*. PhD thesis, University of Oxford, Oxford, United Kingdom, 2010.

[31] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[32] Lionel Heng, Mathias Buerki, Gim Hee Lee, Paul Furgale, Roland Siegwart, and Marc Pollefeys. Infrastructure-based calibration of a multi-camera rig. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[33] Lili Huang and Matthew Barth. A novel multi-planar lidar and computer vision calibration procedure using 2d patterns for automated navigation. In *Intelligent Vehicles Symposium, 2009 IEEE*, pages 117–122. IEEE, 2009.

[34] Peter J Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1):73–101, 1964.

[35] Chr Hug and Aloysius Wehr. Detecting and identifying topographic objects in imaging laser altimeter data. *International archives of photogrammetry and remote sensing*, 32(3 SECT 4W2):19–26, 1997.

[36] Abdallah Kassir and Thierry Peynot. Reliable automatic camera-laser calibration. In *Proceedings of the 2010 Australasian Conference on Robotics & Automation*. ARAA, 2010.

[37] Carl T Kelley. *Iterative methods for optimization*, volume 18. Siam, 1999.

[38] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[39] Rainer Kümmerle, Giorgio Grisetti, Hauke Strasdat, Kurt Konolige, and Wolfram Burgard. g 2 o: A general framework for graph optimization. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 3607–3613. IEEE, 2011.

[40] Quoc V Le and Andrew Y Ng. Joint calibration of multiple sensors. In *IROS*, pages 3651–3658, 2009.

[41] Jesse Levinson and Sebastian Thrun. Automatic online calibration of cameras and lasers. In *Robotics: Science and Systems*, 2013.

[42] Ganhua Li, Yunhui Liu, Li Dong, Xuanping Cai, and Dongxiang Zhou. An algorithm for extrinsic parameters calibration of a camera and a laser range finder

using line features. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3854–3859. IEEE, 2007.

[43] Ming Li, Xin Chen, Xin Li, Bin Ma, and Paul MB Vitányi. The similarity metric. *Information Theory, IEEE Transactions on*, 50(12):3250–3264, 2004.

[44] Tony Lindeberg. Detecting salient blob-like image structures and their scales with a scale-space primal sketch: a method for focus-of-attention. *International Journal of Computer Vision*, 11(3):283–318, 1993.

[45] Chris Linegar, Winston Churchill, and Paul Newman. Work smart, not hard: Recalling relevant experiences for vast-scale but time-constrained localisation. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 90–97. IEEE, 2015.

[46] David G Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999.

[47] Will Maddern, Geoff Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000km: The oxford robotcar dataset. *The International Journal of Robotics Research (IJRR)*, to appear.

[48] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.

[49] Andrew Mastin, Jeremy Kepner, and J Fisher. Automatic registration of lidar and optical images of urban scenes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2639–2646. IEEE, 2009.

[50] N Meierhold. Automatic feature matching between digital images and 2d representations of a 3d laser scanner point cloud. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2010.

[51] Peyman Moghadam, Michael Bosse, and Robert Zlot. Line-based extrinsic calibration of range and image sensors. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 3685–3691. IEEE, 2013.

[52] Ashley Napier, Peter Corke, and Paul Newman. Cross-calibration of push-broom 2d lidars and cameras in natural scenes. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, May 2013.

[53] Oleg Naroditsky, Alexander Patterson, and Kostas Daniilidis. Automatic alignment of a camera with a line scan lidar system. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 3429–3434. IEEE, 2011.

[54] John A Nelder and Roger Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.

[55] Paul Newman, Gabe Sibley, Mike Smith, Mark Cummins, Alastair Harrison, Chris Mei, Ingmar Posner, Robbie Shade, Derik Schroeter, Liz Murphy, Winston Churchill, Dave Cole, and Ian Reid. Navigating, recognizing and describing urban spaces with vision and lasers. *The International Journal of Robotics Research*, 28(11-12):1406–1433, 2009.

[56] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[57] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. [ONLINE]. Available `http://people.csail.mit.edu/torralba/code/spatialenvelope/`.

[58] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001.

[59] Jason Owens, Philip Osteen, and Kostas Daniilidis. Msg-cal: Multi-sensor graph-based calibration. In *International Conference on Robotics and Automation (ICRA), Proc. IEEE*, 2015.

[60] Gaurav Pandey, James McBride, Silvio Savarese, and Ryan Eustice. Extrinsic calibration of a 3d laser scanner and an omnidirectional camera. *IFAC Proceedings Volumes*, 43(16):336–341, 2010.

[61] Gaurav Pandey, James R McBride, Silvio Savarese, and Ryan Eustice. Automatic targetless extrinsic calibration of a 3d lidar and camera by maximizing mutual information. In *AAAI*, 2012.

[62] Gaurav Pandey, James R McBride, Silvio Savarese, and Ryan M Eustice. Automatic extrinsic calibration of vision and lidar by maximizing mutual information. *Journal of Field Robotics*, 32(5):696–722, 2015.

[63] Geoffrey Pascoe, Will Maddern, Alexander D Stewart, and Paul Newman. Farlap: fast robust localisation using appearance priors. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6366–6373. IEEE, 2015.

[64] Anna Petrovskaya and Sebastian Thrun. Model based vehicle detection and tracking for autonomous urban driving. *Autonomous Robots*, 26(2-3):123–139, 2009.

[65] Josien PW Pluim, JB Antoine Maintz, and Max Viergever. Mutual-information-based registration of medical images: a survey. *Medical Imaging, IEEE Transactions on*, 22(8):986–1004, 2003.

[66] Long Quan and Zhongdan Lan. Linear n-point camera pose determination. *IEEE Transactions on pattern analysis and machine intelligence*, 21(8):774–780, 1999.

[67] Carolina Redondo-Cabrera, Roberto J López-Sastre, Javier Acevedo-Rodriguez, and Saturnino Maldonado-Bascón. Surfing the point clouds: Selective 3d spatial pyramids for category-level object recognition. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3458–3465. IEEE, 2012.

[68] Edward Rosten and Tom Drummond. Fusing points and lines for high performance tracking. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pages 1508–1515. IEEE, 2005.

[69] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. IEEE, 2011.

[70] Davide Scaramuzza, Ahad Harati, and Roland Siegwart. Extrinsic self calibration of a camera and a 3d laser range finder from natural scenes. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 4164–4169. IEEE, 2007.

[71] Terry Scott, Akshay A. Morye, Pedro Piniés, Lina M. Paz, Ingmar Posner, and Paul Newman. Exploiting Known Unknowns: Scene Induced Cross-Calibration of Lidar-Stereo Systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015.

[72] Terry Scott, Akshay A. Morye, Pedro Piniés, Lina M. Paz, Ingmar Posner, and Paul Newman. Choosing a time and place for calibration of lidar-camera systems. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Stockholm, Sweden, May 2016.

[73] Claude Elwood Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55, 2001.

[74] Gabe Sibley, Christopher Mei, Ian Reid, and Paul Newman. Vast-scale outdoor navigation using adaptive relative bundle adjustment. *The International Journal of Robotics Research*, 29(8):958–980, 2010.

[75] Aleksandra A Sima and Simon J Buckley. Optimizing sift for matching of short wave infrared and visible wavelength images. *Remote Sensing*, 5(5):2037–2056, 2013.

[76] Alex Stewart. *Localisation using the Appearance of Prior Structure*. PhD thesis, University of Oxford, 2014.

[77] Alex Stewart and Paul Newman. Laps - localisation using appearance of prior structure: 6-dof monocular camera localisation using prior pointclouds. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Minnesota, USA, May 2012.

[78] Stratasys. Objet eden260vs. [ONLINE]. Available `http://www.stratasys.com/3d-printers/design-series/objet-eden260vs`, 2016.

[79] Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 3(Dec):583–617, 2002.

[80] RIEGL Laser Measurement Systems. Riegl vmx-450, 2015.

[81] Zachary Taylor and Juan Nieto. A mutual information approach to automatic calibration of camera and lidar in natural environments. In *Proc. ACRA*, pages 1–8, 2012.

[82] Zachary Taylor and Juan Nieto. Automatic calibration of lidar and camera images using normalized mutual information. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, 2013.

[83] A. Torralba, K.P. Murphy, W.T. Freeman, and M.A. Rubin. Context-based vision system for place and object recognition. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 273–280 vol.1, 2003.

[84] Ranjith Unnikrishnan and Martial Hebert. Fast extrinsic calibration of a laser rangefinder to a camera. In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*. Carnegie Mellon University, 2005.

[85] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. [ONLINE]. Available: `http://www.vlfeat.org/`, 2008.

[86] Dominic Zeng Wang and Ingmar Posner. Voting for voting in online point cloud object detection. *Proceedings of the Robotics: Science and Systems, Rome, Italy*, 1317, 2015.

[87] Dominic Zeng Wang, Ingmar Posner, and Paul Newman. What could move? finding cars, pedestrians and bicyclists in 3d laser data. *International Conference of Robotics and Automation*, 2012.

[88] Ruisheng Wang, Frank P Ferrie, and Jane Macfarlane. Automatic registration of mobile lidar and spherical panoramas. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 33–40. IEEE, 2012.

[89] Yunfeng Wang and Gregory S Chirikjian. Error propagation on the euclidean group with applications to manipulator kinematics. *IEEE Transactions on Robotics*, 22(4):591–602, 2006.

[90] S Wasielewski and Olivier Charles Strauss. Calibration of a multi-sensor system laser rangefinder/camera. In *Intelligent Vehicles' 95 Symposium., Proceedings of the*, pages 472–477. IEEE, 1995.

[91] Ma Weinmann, Mi Weinmann, Stefan Hinz, and Boris Jutzi. Fast and automatic image-based registration of tls data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(6):S62–S70, 2011.

[92] Alan L Yuille, Peter W Hallinan, and David S Cohen. Feature extraction from faces using deformable templates. *International journal of computer vision*, 8(2):99–111, 1992.

[93] Qilong Zhang and Robert Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2301–2306. IEEE, 2004.

[94] Zhengyou Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 666–673. Ieee, 1999.

[95] Huijing Zhao, Yuzhong Chen, and Ryosuke Shibasaki. An efficient extrinsic calibration of a multiple laser scanners and cameras' sensor system on a mobile platform. In *2007 IEEE Intelligent Vehicles Symposium*, pages 422–427. IEEE, 2007.

[96] Yipu Zhao, Yuanfang Wang, and Yichang Tsai. 2d-image to 3d-range registration in urban environments via scene categorization and combination of similarity measurements. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1866–1872. IEEE, 2016.